

TECHNICAL UNIVERSITY OF MOLDOVA



As manuscript
U.D.C: 616-089.5-083.98:004.8

IAPĂSCURTĂ VICTOR

**INTELLIGENT METHODS AND SOFTWARE
ALGORITHMS FOR INTENSIVE CARE. A COMPLEXITY
SCIENCE APPROACH**

**122.03—MODELING, MATHEMATICAL METHODS, SOFTWARE PRODUCTS
321.19 – ANESTHESIA AND INTENSIVE CARE**

Doctor habilitated thesis in computer science and medical science

Scientific consultants:

Bostan Viorel
Ph.D. habil. in engineering science,
cm ASM

Belii Adrian
Ph.D. habil. in medical science

Author:

Iapăscurtă Victor, Ph.D.

Chișinău, 2026

UNIVERSITATEA TEHNICĂ A MOLDOVEI



Cu titlu de manuscris
C.Z.U: 616-089.5-083.98:004.8

IAPĂSCURTĂ VICTOR

**METODE ȘI ALGORITMI SOFTWARE INTELIGENȚI
PENTRU TERAPIA INTENSIVĂ. ABORDARE PRIN PRISMA
ȘTIINȚEI COMPLEXITĂȚII**

**122.03 – MODELARE, METODE MATEMATICE, PRODUSE PROGRAM
321.19 – ANESTEZIOLOGIE ȘI TERAPIE INTENSIVĂ**

Teză de doctor habilitat în informatică și științe medicale

Consultanți științifici:

Bostan Viorel
dr. hab. șt. ingineresci
mc AȘM

Belii Adrian
dr. hab. șt. med.

Autorul:

Iapăscurtă Victor
dr. șt. med.

Chișinău, 2026

CONTENTS

SUMMARY (Rom)	8
SUMMARY (Eng)	9
LIST OF TABLES	10
LIST OF FIGURES	12
LIST OF ABBREVIATIONS	17
INTRODUCTION	20
1. CONCEPTUAL FRAMEWORK - A COMPLEXITY SCIENCE APPROACH TO INTELLIGENT METHODS IN INTENSIVE CARE	28
1.1. Background	28
1.1.1. Historical Development of Complexity Science	29
1.1.2. Key Concepts in Complexity Science	30
1.1.3. Distinctions with Systems Theory	31
1.1.4. Applications of Complexity Science in General and in Healthcare	32
1.2. Complexity Science Interactions with Agent-Based Modeling (ABM), System Dynamics Modeling (SDM), and Hybrid Approaches	33
1.3. The Interplay of Complexity Science and Machine Learning: A Common View	33
1.4. Argumentation for Sepsis and Epilepsy as Central Pathological States in Intensive Care	34
1.5. Complexity Data Science: Reframing Intelligent Methods in Intensive Care	36
1.6. Synthesis of Chapter 1	39
2. AGENT-BASED MODELING IN INTENSIVE CARE: A COMPLEXITY SCIENCE APPROACH	42
2.1. Agent-based modeling in medicine and complexity science	42
2.1.1. Evolution of Complexity Science in Medicine	42
2.1.2. Agent-Based Modeling: A Computational Procedure	43
2.1.3. Intensive Care Applications	43
2.1.4. Difficulties and Prospects	44
2.2. Mathematical Model of Agent-Based Modeling	44
2.2.1. Core Components of ABM	44
2.2.2. Mathematical Modeling	45
2.3. Application to Critical Care	46
2.4. Agent-Based Modeling of Cardiogenic Pulmonary Edema: Model Description	46
2.4.1. Model Structure	47
2.4.2. Dynamics of the Model	48
2.4.3. Implementation of the Cardiogenic Pulmonary Edema Model	48
2.4.4. Simulation Dynamics	49
2.4.5. Technical Problems	49
2.5. Cardiogenic Pulmonary Edema Model Results	49
2.5.1. Normal Scenario	50
2.5.2. CPE Scenario	51

2.5.3.	Parameter Sensitivity		51
2.5.4.	Further Remarks		52
2.6.	Cardiogenic Pulmonary Edema Model: Clinical and Educational Potential, Limitations, and Future Directions		53
2.6.1.	Applications in Clinical Medicine		53
2.6.2.	Educational Value		54
2.6.3.	Limitations and Future Research		54
2.7.	Synthesis of Chapter 2.		55
3.	SYSTEM DYNAMICS MODELING IN COMPLEXITY SCIENCE: APPLICATIONS TO INTENSIVE CARE MEDICINE (HUMAN CIRCULATORY SYSTEM)		57
3.1.	Complexity Science and System Theory in Medicine (ICU). A System Dynamics Approach		57
3.2.	General Mathematical Formalism Behind System Dynamics Modeling		60
3.2.1.	Basic Components and Mathematical Aspects of SDM in the Circulatory System		61
3.3.	Real-Time Data Integration into SDM and Model Generation		63
3.4.	Simulation results and analysis in MI		68
3.5.	Discussion and Future Directions for SDM Circulatory System Modeling		70
3.6.	Synthesis of Chapter 3		71
4.	HYBRID MODELING IN INTENSIVE CARE: COMBINING SYSTEM DYNAMICS AND AGENT-BASED MODELING TO IMPROVE PHYSIOLOGICAL SIMULATIONS		74
4.1.	Hybrid Modeling in Medicine and Intensive Care		74
4.2.	Model framework: Combining SDM and ABM for Simulating Maternal-Placental-Fetal Interactions		76
4.3.	Mathematical and Computational Foundations of the Hybrid Model		79
4.4.	Scenario-Based Simulations: Pathophysiological Dynamics		80
4.5.	Applications in Clinical Practice, Educational Contexts, Limitations, and Future Development		83
4.6.	Digital Twins in Intensive Care		84
4.7.	Synthesis of Chapter 4		85
5.	FROM TRADITIONAL MEDICINE TO COMPUTATIONAL PARADIGMS: COMPLEXITY-DRIVEN APPROACHES TO DIAGNOSIS AND PROGNOSIS IN CRITICAL CONDITIONS		89
5.1.	Medical diagnosis and prognosis: tracing the evolution from ancient times to the present era		89
5.1.1.	Historical perspective on the concept evolution		89
5.1.2.	Logical and Mathematical Aspects of Medical Diagnosis in Complexity Science		90
5.1.3.	The diagnosis versus prognosis problem		93
5.1.4.	Medical aspects of the epilepsy and sepsis research field		96
5.2.	Aspects of representation learning, complexity science, and machine learning relevant to the research topic areas		98

5.2.1.	Representation learning as a lens for current research	98
5.2.2.	Contemporary research from the point of complexity science	99
5.2.3.	Machine learning techniques applied in medicine and medical research	100
5.3.	Medical prognosis using machine learning in epilepsy and sepsis	101
5.3.1.	Representation learning and its application in research	101
5.3.2.	Methods and algorithms used for classification purposes	101
5.3.3.	State-of-the-art examples and open research issues	107
5.4.	Synthesis of Chapter 5	109
6.	FROM RAW DATA TO ACTIONABLE INSIGHTS: IMPUTATION, COMPLEXITY METRICS, AND LEARNING REPRESENTATIONS FOR CRITICAL CARE APPLICATIONS	112
6.1.	Data sets used in the present research	112
6.1.1.	An overview of the data utilized in this part of the study	112
6.1.2.	Preliminary data processing. Data gap filling	115
6.2.	Concepts and instruments of algorithmic complexity during the preprocessing phase	123
6.2.1	The Kolmogorov-Chaitin complexity and the Block Decomposition Method as fundamental methodologies for determining algorithmic complexity	123
6.2.2.	Assessment of the Kolmogorov-Chaitin complexity concerning epilepsy dataset information	124
6.2.3.	Assessment of the Kolmogorov-Chaitin complexity regarding sepsis dataset	129
6.3.	Utilization of topological representation learning techniques for research data	130
6.3.1.	Text representation learning. Word embedding	130
6.3.2.	Representation learning and application of NLP tools to transformed (textual) EEG data	136
6.3.3.	Representation learning on sepsis data	141
6.4.	Intermediary results and stage concluding remarks	147
6.5.	Synthesis of Chapter 6	150
7.	INTEGRATING ALGORITHMIC COMPLEXITY IN ML FOR INTENSIVE CARE APPLICATIONS: FINDINGS, DISCUSSIONS, AND PRACTICAL IMPLICATIONS	153
7.1.	Stages of machine learning system development. Techniques and algorithms used	153
7.1.1.	Main elements of a machine learning cycle	153
7.1.2.	Governing aspects of ML for medical purposes	154
7.1.3.	Criteria used to evaluate the performance of machine learning model	155
7.2.	Training, testing, validation and comparison of developed systems	156
7.2.1.	1. Algorithmic complexity of the EEG time series (numeric representation) for exploratory data investigation	156
7.2.2.	Using modified EEG data for machine learning	157
7.2.3.	Using the sepsis dataset for machine learning applications	166

7.3.	Explainability/Interpretability aspects of the created models	168
7.3.1.	Models using EEG data	168
7.3.2.	Models based on sepsis data	171
7.4.	Considerations regarding the practical application of generated models	173
7.5.	Software applications developed using established models	175
7.6.	Concluding discussion and future steps	177
7.7.	Synthesis of Chapter 7	181
8	LARGE LANGUAGE MODELS AND MULTI-AGENT SYSTEMS IN INTENSIVE CARE: SEPSIS MANAGEMENT IMPROVEMENT	184
8.1.	Foundations of Large Language Models, Retrieval-Augmented Generation, and Multi-Agent Systems in Intensive Care	184
8.2.	Using Retrieval-Augmented Generation for Domain Specific Tasks	187
8.2.1.	Case Study: Retrieval Augmented Generation for Domain-Specific Question Answering	187
8.2.1.1.	Background and Study Context	187
8.2.1.2.	Problem Definition	187
8.2.1.3.	Methods and Implementation	188
8.2.1.4.	Study Results	191
8.2.1.5.	Discussion and Takeaways	192
8.3.	RAG in ICU	192
8.4.	Multi-Agent AI Systems for Sepsis Care in ICUs	193
8.4.1.	Case Study: Proof-of-Concept Multi-Agent System for Early Sepsis Management in Community-Acquired Pneumonia	193
8.4.1.1.	Problem Definition	193
8.4.1.2.	Methods	193
8.4.1.3.	Results	195
8.4.1.4.	Discussion and Lessons Learned	197
8.4.2.	Follow-up Case Study: Scaling Multi-Agent System to Multiple Sepsis Cases and Hallucination Mitigation	197
8.4.2.1.	Problem Statement	197
8.4.2.2.	Methods	197
8.4.2.3.	Results	198
8.4.2.4.	Discussion and Lessons Learned	203
8.5.	Synthesis of Chapter 8	204
	GENERAL CONCLUSIONS	206
	RECOMMENDATIONS	210
	BIBLIOGRAPHY	213
	SUPPLEMENTARY MATERIALS	238
Appendix 1	Reference letter concerning the Algorithmic Information Dynamics Project	238
Appendix 2	Non-linear regression methods for missing values imputation	239
Appendix 3	General description of the architecture of a BERT model built in the study, and the BERT classifier	244
Appendix 4	Graphical user interface (GUI) and code for additional software applications built in the study	248

Appendix 5	Patents and certificates for inventions that formalize the study results		265
	DISCLAIMER OF LIABILITY		271
	CURRICULUM VITAE		272

ADNOTARE

Iapăscurtă Victor, “Metode și algoritmi software inteligenți pentru terapia intensivă. Abordare prin prisma științei complexității”, teza de doctor habilitat în informatică și medicină, Chișinău, 2026

Structura tezei. Teza cuprinde 275 de pagini (introducere, 8 capitole, concluzii), o bibliografie cu 401 de surse, 5 anexe, 87 de figuri și 37 de tabele, în baza a 54 de lucrări publicate, inclusiv 25 de articole, dintre care 14 în reviste/ediții cu factor de impact. **Cuvinte cheie:** știința complexității, terapie intensivă, sepsis, epilepsie, modelare bazată pe agenți, modelare dinamică a sistemelor, învățare automată, învățare a reprezentării, complexitate Kolmogorov-Chaitin, modele de limbaj mari, sisteme multi-agent, generare augmentată prin recuperare, gemeni digitali. **Scopul studiului:** Dezvoltarea unor noi metode inteligente, algoritmi și aplicații software pentru medicina de terapie intensivă, bazate pe știința complexității, în vederea îmbunătățirii diagnosticului precoce, a predicției și a gestionării afecțiunilor grave, precum sepsisul și crizele epileptice.

Obiectivele studiului: Stabilirea unui cadru de știință a complexității pentru sistemele de terapie intensivă; Evaluarea modelării bazate pe agenți, dinamica sistemelor și hibridă pentru simulări fiziologice; Identificarea și analiza seturilor mari de date pentru sepsis și epilepsie; Dezvoltarea de preprocesare avansată a datelor cu estimare algoritmică a complexității; Integrarea învățării reprezentării și a învățării automate pentru predicții îmbunătățite; Încorporarea sistemelor multi-agent bazate pe modele de limbaj mari pentru gestionarea sepsisului; și Crearea de instrumente software pentru aplicații clinice și de cercetare.

Noutate științifică și originalitate. Teza introduce o integrare nouă a științei complexității cu terapia intensivă, valorificând neliniaritatea și emergența pentru a aborda lacunele de diagnostic. Este pionieră în utilizarea complexității Kolmogorov-Chaitin (KCC) prin descompunerea în blocuri pentru preprocesarea semnalelor biomedicale, îmbunătățind detectarea tiparelor în datele nestăționare. Cadrul de modelare hibridă, care combină abordări bazate pe agenți și dinamica sistemului pentru simulări, cum ar fi livrarea oxigenului fetal, este inovator. În plus, aplicarea modelelor de limbaj mari cu generare augmentată prin recuperare și sisteme multi-agent pentru gestionarea sepsisului reprezintă o abordare de ultimă generație a deciziilor clinice în timp real, distingând această lucrare de metodele reduționiste tradiționale.

Principalele rezultate. Cercetarea stabilește un cadru științific al complexității, modelând cu succes procese fiziologice precum edemul pulmonar cardiogen și livrarea oxigenului fetal, utilizând metode bazate pe agenți, dinamica sistemului și hibride. Preprocesarea avansată, inclusiv estimarea KCC, transformă datele EEG și sepsis pentru învățarea automată, atingând performanțe predictive ridicate. Instrumente software create precum ISAAC-EEG-BDM-NLP și ISAAC Sepsis Expert permit detectarea și gestionarea precoce, sistemele bazate pe modele de limbaj mari îmbunătățind respectarea ghidurilor privind sepsisul.

Semnificația teoretică a cercetării. Teza promovează știința complexității prin formalizarea aplicării sale în terapia intensivă, integrând-o cu modelare, învățare automată și inteligență artificială. Aceasta contribuie cu o nouă paradigmă a științei datelor despre complexitate, conectând teoria sistemelor și abordările bazate pe date. Utilizarea KCC pentru analiza semnalelor biomedicale oferă un salt teoretic în captarea dinamicii emergente, în timp ce modelarea hibridă și sistemele multi-agent oferă noi cadre pentru înțelegerea fenomenelor comune pentru terapia intensivă, îmbogățind cercetarea transdisciplinară.

Valoarea aplicativă a lucrării. Instrumentele dezvoltate oferă soluții practice pentru predicția și gestionarea timpurie a sepsisului și prognoza convulsiilor bazată pe EEG, îmbunătățind luarea deciziilor clinice și îngrijirea personalizată. Aceste aplicații sunt scalabile pentru medii cu resurse limitate, cum ar fi Moldova, sprijinind triajul critic, planificarea intervențiilor și educația medicală prin simulări interactive.

Implementarea rezultatelor științifice. Prototipurile software, ISAAC-EEG-BDM-NLP pentru cercetarea epilepsiei și ISAAC Sepsis Expert și MA System pentru gestionarea sepsisului clinic, sunt concepute pentru integrarea în sistemul de sănătate din Republica Moldova. Sunt planificate implementări pilot în unitățile de terapie intensivă, în așteptarea aprobării autorităților de reglementare și a validării prospective, cu potențialul unei adoptări pe scară largă prin intermediul platformelor bazate pe cloud pentru a îmbunătăți rezultatele pacienților.

SUMMARY

Iapăscuță Victor, "Intelligent methods and software algorithms for intensive care. A complexity science approach," postdoctoral thesis in computer sciences and medicine, Chisinau, 2026

Thesis structure. The thesis has 275 pages (introduction, 8 chapters, conclusions), bibliography of 401 sources, 5 appendices, 87 figures and 37 tables, based on 54 published papers, including 25 articles, impact factor journal/edition with 14 publications.

Keywords: complexity science, intensive care medicine, sepsis, epilepsy, agent-based modeling, system dynamics modeling, machine learning, representation learning, Kolmogorov-Chaitin complexity, large language models, multi-agent systems, retrieval-augmented generation, data preprocessing, digital twins.

The aim of the study. To develop new intelligent methods, algorithms and software applications for intensive care medicine based on complexity science to improve early diagnosis, prediction and management of severe conditions such as sepsis and epileptic seizures.

Objectives of the study: establishing a complexity science framework for intensive care systems; evaluating agent-based, system dynamics, and hybrid modeling for physiological simulations; identifying and analyzing datasets for sepsis and epilepsy; developing advanced data preprocessing with algorithmic complexity estimation; integrating representation learning and machine learning for improved predictions; incorporating large language model-based multi-agent systems for sepsis management; and creating software tools for clinical and research applications in Moldova.

Scientific novelty and originality. The thesis introduces a novel integration of complexity science into intensive care to exploit nonlinearity and emergence to fill diagnostic gaps. It is the first to use KCC (block decomposition) as preprocessing method for biomedical signals to detect patterns in non-stationary data. Hybrid modelling framework combining agent-based and system dynamics approach for multiscale simulations such as fetal oxygen delivery is introduced. Large language models with retrieval augmented generation and multi-agent systems for sepsis management are new approaches to real-time clinical decision making which set this work apart from traditional reductionist methods.

Main results. The thesis develops complexity science framework that successfully models physiological processes (cardiogenic pulmonary edema and fetal oxygen delivery) using agent-based, system dynamics and hybrid methods. KCC estimation and imputation preprocessing transforms EEG and sepsis data for machine learning with high predictive performance. Software tools ISAAC-EEG-BDM-NLP and ISAAC Sepsis Expert allow early detection and management of sepsis while large language model based systems improve sepsis guideline adherence.

Theoretical contribution of the work. The thesis formalizes complexity science application in intensive care and combines it with modeling, machine learning and AI; presents a new complexity data science paradigm linking systems theory and data driven approaches; uses KCC for biomedical signal analysis which is a theoretical leap to capture emergent dynamics; hybrid modeling and multi-agent systems introduce new frameworks for multiscale ICU phenomena enriching transdisciplinary research.

Applicative value of the work. The developed tools provide practical solutions for early sepsis prediction and management and EEG seizure forecasting; enhance clinical decision making and personalized care. These applications are scalable for resource poor settings (Moldova) ICU triage, intervention planning and medical education through simulations.

Implementation of scientific results. The software prototypes, ISAAC-EEG-BDM-NLP for epilepsy research and ISAAC Sepsis Expert and MA System for clinical sepsis management, are designed to be integrated into Moldova's healthcare system. Pilot implementations in ICUs will be done once regulatory approval and prospective validation are obtained and then cloud based platforms can be used to scale up adoption to improve patient outcomes.

LIST OF TABLES

1.	Table 1.1. System theory & Complexity science: similarities	32
2.	Table 3.1. Initial conditions for the SDM model	66
3.	Table 3.2. A minor excerpt of the simulation findings	68
4.	Table 5.1. Discoveries/developments related to the current research	89
5.	Table 5.2. A summary of publications on high-performance EEG classification and prediction systems (2020–2025)	108
6.	Table 5.3. High-performance systems for sepsis prediction (2020-2025)	109
7.	Table 6.1. Overview of the datasets employed in the study.	114
8.	Table 6.2. An excerpt from an original sepsis subset	117
9.	Table 6.3. Relations between data as coefficients of correlation	122
10.	Table 6.4. Coefficients of linear regression for the parameters of significance.	123
11.	Table 6.5. An illustration of recovered sepsis information	123
12.	Table 6.6. Unprocessed EEG data	125
13.	Table 6.7. Data that has been binarized according to the mean threshold value per row	126
14.	Table 6.8. Algorithmic complexity calculated via BDM	126
15.	Table 6.9. Preliminary sepsis data (HR, SBP, and DBP), organized in a 3 x 3 array	130
16.	Table 6.10. Binarized information (employing the threshold value on a per-row basis)	130
17.	Table 6.11. Algorithmic complexity derived from BDM	130
18.	Table 6.12. The format of the sepsis data used for machine learning	130
19.	Table 6.13. Data sets used for building Word2Vec models	138
20.	Table 6.14. Data sets that were utilized in constructing the BERT models	140
21.	Table 6.15. Performance of a GBM classifier on PCA-transformed data after re-attaching original labels	144
22.	Table 6.16. Confusion matrices for LDA on the sepsis test set	145
23.	Table 6.17. The confusion matrices and performance metrics for three distinct data recovery techniques	149

24.	Table 7.1. Data for machine learning using TF-IDF		157
25.	Table 7.2. Information for machine learning from NLP models		158
26.	Table 7.3. Variables utilized in the development of the Word2Vec model		159
27.	Table 7.4. The most effective NLP models built with dog 2 EEG data		165
28.	Table 7.5. Best main hyperparameters: 4 ML algorithms/model		166
29.	Table 7.6. Performance of the 4 ML algorithms/models.		167
30.	Table 7.7. Comparison of several sepsis forecasting systems		168
31.	Table 7.8. SHAP values for two samples		173
32.	Table 7.9. Comparative performance of models developed on the same EEG data set		178
33.	Table 7.10. High-performance systems for sepsis prediction (last 5 years)		179
34.	Table 8.1. The performance of the three RAG variants		191
35.	Table 8.2. Clinical vignette used for the pneumonia sepsis case		195
36.	Table 8.3. Input vignettes and generated outputs for selected sepsis cases		199
37.	Table 8.4. TruLens evaluation metrics (GPT-3.5 Turbo) across eight MIMIC-IV sepsis cases		218

LIST OF FIGURES

1.	Figure 1.1. The count of publications in PubMed: complexity science in association with intensive care	28
2.	Figure 1.2. The weight of key concepts in percent for PubMed searches: complexity science and systems theory (exact matches for the respective terms)	31
3.	Figure 1.3. A holistic curriculum for CDS	37
4.	Figure 2.1. General view of the model interface in the “Normal” scenario	47
5.	Figure 2.2. Plot view in “Normal” case	50
6.	Figure 2.3. Plot view in the “Cardiogenic-Pulmonary-Edema” scenario	51
7.	Figure 2.4. Dynamics of extravascular water volume over time for varying HP levels	52
8.	Figure 3.1. The system dynamic scheme for the created model	65
9.	Figure 3.2. User interface of the model	68
10.	Figure 3.3. PV loops in myocardial infarction	69
11.	Figure 3.4. Temporal dynamics of Kolmogorov-Chaitin complexity in a simulated scenario of myocardial infarction	70
12.	Figure 4.1. A complete view of the prototype interface	77
13.	Figure 4.2. The model's system dynamic scheme: (1) the circulatory systems of the mother, (2) the fetus, and (3) the placenta	78
14.	Figure 4.3. The green area and the lower display show the oxygen supply to the fetus (standard conditions a) and maternal iron deficiency (b)	81
15.	Figure 4.4. The fetus oxygen delivery (lower monitor and green area) in (a) Abruptio Placentae scenario and (b) typical uterine contraction	82
16.	Figure 4.5. The green region and lower monitor indicate the fetus's oxygen delivery: Scenarios of (a) complex pathology and (b) anesthesia. Green region and lower monitor show oxygen delivery to the fetus in (a) complex pathology and (b) anesthesia	82
17.	Figure 4.6. Oxygen delivery to the fetus in abruptio placentae (red color line) and Kolmogorov-Chaitin complexity	83
18.	Figure 5.1. Conceptual meaning of AI-based predictive analytics in diagnosing and predicting the progression of diseases	94

19	Figure 5.2. Fitting a line to data with linear regression	103
20	Figure 5.3. Partition of the feature space by lines parallel to coordinate axes	103
21	Figure 5.4. A fully connected (FC) artificial neural network (ANN) with four predictors and two hidden layers	104
22	Figure 5.5. Schematic of a basic RNN	106
23	Figure 5.6. Support vector machine (SVM)	106
24	Figure 6.1. Visual representation of EEG data from ten electrodes (channels)	113
25	Figure 6.2. Visual representation of the raw sepsis data	114
26	Figure 6.3. Data processing sequence in the present study (a CONSORT-style diagram)	121
27	Figure 6.4. Temporal series describing KCC values for a 10-minute EEG segment	126
28	Figure 6.5. The predominant KCC values (converted to words) associated with every topic	127
29	Figure 6.6. Distinct values for topic 1 and topic 2 BDM	128
30	Figure 6.7. The distinction between the topics illustrated by box plots	129
31	Figure 6.8 The Continuous Bag of Words & Skip-gram Models	133
32	Figure 6.9. The architectural framework of fastText for a sentence that has n-gram attributes	134
33	Figure 6.10. Diagrammatic depiction of the configuration of a Long Short-Term Memory (LSTM) network	135
34	Figure 6.11. A scatter diagram representing two samples extracted from the arithmetic examination dataset	136
35	Figure 6.12. Locally Linear Embedding (LLE) applied to arithmetic-test data	137
36	Figure 6.13. Multidimensional Scaling (MDS) (a) and Isomap (b) on arithmetic-test data	137
37	Figure 6.14t-SNE (a) and spectral embedding (b) on arithmetic-test data	137
38	Figure 6.15. UMAP diagnostic visualizations for Dog 2	139
39	Figure 6.16. PCA of 30 algorithmic-complexity terms (two principal components)	139

40	Figure 6.17. BiLSTM-based classifier schematic	141
41	Figure 6.18. The biplot of the first two principal components	142
42	Figure 6.19. Individual and cumulative explained variance for the sepsis dataset	142
43	Figure 6.20. Example decision tree predicting patient body temperature using principal components 3 and 10	143
44	Figure 6.21. LDA plots for the sepsis training set using two features	144
45	Figure 6.22. PCA and LDA plots of the sepsis dataset	145
46	Figure 6.23. 3D representation of three sepsis variables	146
47	Figure 6.24. LLE (a) and modified LLE (b) on sepsis data	146
48	Figure 6.25. MDS (a) and Isomap (b) on sepsis data	147
49	Figure 6.26. t-SNE (a) and spectral embedding (b) on sepsis data	147
50	Figure 6.27 Classification performance, measured by the Area Under the Curve (AUC), of three GBM models	148
51	Figure 6.28. Statistical evaluation of the efficacy of three machine learning models	149
52	Figure 7.1. Main elements and steps in the creation of a machine learning system	153
53	Figure 7.2. Partitioning of data into training and test sets	153
54	Figure 7.3. Aggregated AC over time: (a) for an “interictal”/normal EEG clip, and (b) for a “preictal”/pre-epileptic clip.	157
55	Figure 7.4. The ROC curve (a) and the contingency table (b) of the trained model	159
56	Figure 7.5. The efficacy measured by AUC (a) and Precision-Recall AUC (b) of Skip Gram Word2vec model on Dog 1 test	160
57	Figure 7.6. The efficacy measured by AUC (a) and Precision-Recall AUC (b) of a Skip Gram Word2vec model on Dog 2 test	160
58	Figure 7.7. Comparison (c) between CBOW (a) and Skip-Gram (b) models	160
59	Figure 7.8. The performance of a fastText model on EEG data set	161
60	Figure 7.9. The efficacy measured by AUC (a) and Precision-Recall AUC (b) of a BERT model on the Dog 2 test	162
61	Figure 7.10. Performance (by AUC) of a BERT model on the Dog	162

	2 test set	
62	Figure 7.11. Visualizations of the BERT model built on the Dog 2 test set	163
63	Figure 7.12. Visualization of a particular layer (#2) and head (#0) for a particular pair of tokens (def → cc) as heatmaps	163
64	Figure 7.13. Loss of the PyTorch BiLSTM during training with Dog 2 data	164
65	Figure 7.14. Training and validation results of the Keras model with Dog 2 data	164
66	Figure 7.15. Receiver Operating Characteristic curves for four different algorithms (DNN, GBM, GLM, DRF)	167
67	Figure 7.16. (a) An original “pretest” image; (b) and (c) the heatmap for one of the first layers and 9th layer (using VGG-16 model) when input into a CNN for classification	170
68	Figure 7.17. An original “during-the-test” image; (b) and (c) the heatmap for one of the first layers and 9th layer (using VGG-16 model) when input into a CNN for classification	170
69	Figure 7.18. Heat maps of AC (algorithmic complexity) for six instances of AC time series from different categories	171
70	Figure 7.19. Distribution density graphs showing allocation of AC levels	171
71	Figure 7.20. Partial dependence visualization	172
72	Figure 7.21. A SHAP-summary plot generated with Wolfram Engine 13.3 for one of the GBM models built on sepsis data	173
73	Figure 7.22. Graphical User Interface (GUI) to analyze the algorithmic complexity of EEG using ISAAC-EEG-BDM-NLP/Word2vec_2.1.0 software	175
74	Figure 7.23. GUI of software application for early prediction of sepsis (ISAAC Sepsis Expert - 2.1.2)	177
75	Figure 8.1. Architecture of the created RAG system	188
76	Figure 8.2. Process of building the vector database	188
77	Figure 8.3. Main steps of the RAG data pipeline	189
78	Figure 8.4. Query embedding proximity to retrieved chunks for relevant questions	189
79	Figure 8.5. Query embedding and spread of retrieved chunks for an irrelevant query	190

80	Figure 8.6. Evaluation of context relevance using Chain-of-Thought reasoning		190
81	Figure 8.7. Evaluation of groundedness using Chain-of-Thought reasoning		191
82	Figure 8.8. Evaluation of answer relevance		191
83	Figure 8.9. RAG component in the multi-agent sepsis system		194
84	Figure 8.10. Overall architecture and data flow of the multi-agent system		194
85	Figure 8.11. Recommendations generated by the MA system		196
86	Figure 8.12. TruLens evaluation results for the single-case proof-of-concept		197
87	Figure 8.13. Individual agent architecture and internal data flows		198
88	Figure 8.14. Comparison of programmatic (TruLens) and human expert evaluation scores		198
89	Figure 8.15. Summary of hallucination and excessive-information problems in recommendations for sepsis across three LLMs		203

LIST OF ABBREVIATIONS

ABM	- Agent-based modeling
AI	- Artificial intelligence
AI Crew	Platform for multi-agent systems orchestration
AID	- Algorithmic Information Dynamics
ANN	- Artificial neural network
AP	- Algorithmic probability
AUC ROC	- Area under the ROC curve
AutoML	- Automated machine learning
BDM	- Block decomposition method
BN (NB)	- Bayesian network/Naive Bayes
CAS	- Complex adaptive system
CCA	- Complete case analysis
CI	- Confidence interval
CNN	- Convolutional neural network
ChromaDB	- Software for vector store creation (with indexing)
CS/CDS	- Complexity science/complexity data science
DBP	- Diastolic blood pressure
DRF	- Distributed random forest
DT	- Digital twin
EDA	- Exploratory data analysis
EHR	- Electronic health records
FDA	- Food and Drug Administration (USA)
GBM	- Gradient boosting machine
GDPR	- General Data Protection Regulation (EU)
GLM	- Generalized linear model
HIPAA	- Health Insurance Portability and Accountability Act (USA)
HR	- Heart rate
H2O.ai	- Platform for machine learning
ICU / ICM	- Intensive care unit / intensive care medicine
Isomap	- Isometric mapping
K or KCC	- Kolmogorov-Chaitin complexity
LIME	Local Interpretable Model-agnostic Explanations
LinR	- Linear regression
LLE	- Locally linear embedding

LLM	- Large language model
LOCF	- Last observation carried forward
LogR	- Logistic regression
LPM	- Lumped parameter model
MA	- Multi-agent (system)
MDS	- Multi-dimensional scaling
ML	- Machine Learning
MLS/AIS	- Machine learning system / Artificial intelligence system
NetLogo	- Programming language and development environment
ODE	- Ordinary differential equation
PD	- Partial dependence (plot)
PROBAST	- Prediction model Risk Of Bias ASsessment Tool
PR-curve	- Precision-recall curve
Python	- Programming language and development environment
R	- Programming language and development environment
RAG	- Retrieval-augmented generation
RL	Representation learning
RNN	- Recurrent neural network
RR	- Respiratory rate
SBP	- Systolic blood pressure
SDM	- System dynamics modeling
SE	- Spectral embedding
SHAP	- SHapley Additive exPlanations (plot)
SOFA	- Sequential Organ Failure Assessment (score)
ST	- System theory
STARD	- Standards for Reporting of Diagnostic Accuracy Study
SVM	- Support vector machine
Temp	- (body) temperature
TRIPOD	- Transparent Reporting of a multivariable prediction model for Individual Prognosis or Diagnosis
t-SNE	- t-Stochastic neighbor embedding
UMAP	- Uniform Manifold Approximation and Projection
Wolfram	- Programming language and development environment
XAI	- E(X)plainable Artificial Intelligence

© Iapăscurtă Victor, 2026

INTRODUCTION

In the high-pressure context of Intensive Care Units (ICUs), where patient outcomes hinge on the functioning of physiological systems, medical interventions, and the collaborative efforts of healthcare teams, simplistic reductionist approaches often fall short. Reductionism tends to dissect complex phenomena into separate parts, neglecting the emergent behaviors and nonlinear dynamics that are prevalent in critical conditions such as sepsis and epilepsy. This thesis advocates for a paradigm shift by employing complexity science, a transdisciplinary field that examines systems characterized by numerous interacting components, resulting in holistic properties that emerge unpredictably from localized interactions (Wiener, 1948). By conceptualizing ICU challenges as complex adaptive systems, we aim to create sophisticated tools for diagnosis, prognosis, and management in critical care that bridge existing gaps in medical research and practice.

Complexity science serves as an effective framework for addressing ICU issues due to its origins in late 19th-century nonlinear dynamics and its formalization through mid-20th-century cybernetics (Wiener, 1948) and information theory (Shannon, 1948). The chaos theory introduced in the 1960s highlighted sensitivity to initial conditions (Lorenz, 1963), while dissipative structures within nonequilibrium systems (Prigogine, 1977) and concepts of self-organization have led to significant advances in big data applications and artificial intelligence. Unlike systems theory—which focuses on equilibrium states and structural hierarchies (von Bertalanffy, 1950s)—complexity science emphasizes adaptation, emergence, and nonequilibrium states. Therefore, it is particularly well-suited for modeling unpredictable events like septic shock or status epilepticus (Lehnetz, 2008).

Recent trends observed on PubMed indicate growth within this field; however, it has yet to be fully integrated into intensive care practices (Braithwaite et al., 2018). This thesis examines sepsis and epilepsy as illustrative case studies. Sepsis stands out as a leading cause of mortality within ICUs—characterized by nonlinear escalation—and represents a critical state exhibiting power law behavior (Mann-Salinas et al., 2013). Epilepsy can be investigated through multivariate time series analysis alongside neuronal dynamics (Sohanian, Haghighi & Markazi, 2017).

Our contributions synthesize principles from complexity across various modeling techniques alongside data processing methods involving machine learning (ML) and AI. For instance, “complexity data science” [Emmert-Streib et al., 2024] quantifies nonlinearity and emergence found within high-dimensional ICU datasets while developing interpretable predictive algorithms aimed at facilitating early interventions.

Ultimately, the transdisciplinary tools developed through this research endeavor target reducing ICU mortality rates while optimizing resource allocation and paving avenues toward personalized proactive critical care.

Research Motivation. Patient outcomes within intensive care settings face immense challenges since delays in identifying sepsis or epileptic seizures can result in irreversible harm. In clinical environments I have observed that conventional monitoring tools often overlook subtle physiological alterations; thus, complexity science was examined as a potential framework to capture the nonlinearity and emergent characteristics inherent in these conditions. The integration of advanced computational methodologies—such as agent-based modeling (ABM), system dynamics modeling (SDM), and hybrid models—holds promise for simulating complex scenarios encountered within ICUs when paired with ML techniques.

Successful applications of ML across various domains inspired our pursuit of utilizing ML for biomedical signal analysis—a novel implementation utilizing Kolmogorov-Chaitin complexity—and crafting predictive systems that exceed existing benchmarks found within literature (AUC >0.95 for sepsis). Moreover, efforts aimed at developing software solutions tailored for Moldova's healthcare system—where resource constraints limit access to advanced diagnostic capabilities—were also influential: creating scalable solutions such as ISAAC-EEG-BDM-NLP or ISAAC Sepsis Expert enables clinicians and researchers alike to maximize efficacy despite limited resources; thereby fostering precision medicine which may alleviate both clinical burdens and economic strains associated with critical illnesses.

Research Field. The focus of this thesis lies at the intersection of informatics/medical informatics, biomedical engineering, and complexity science specifically applied through intensive care intelligence methodologies. We employ agent-based modeling along with system dynamics modeling techniques coupled with machine learning methods including representation learning strategies based on algorithmic complexity metrics such as Kolmogorov-Chaitin complexity. These approaches will facilitate advancements concerning early diagnosis as well as management strategies targeting sepsis and epilepsy ultimately enhancing decision-making processes both clinically oriented research environments.

Timeliness and Importance of the Problem Addressed. The intricate nature of intensive care medicine indicates an urgent need for innovative strategies aimed at enhancing early detection capabilities concerning critical illnesses like sepsis or epilepsy which contribute significantly towards elevated mortality rates globally within ICUs. These dynamic nonlinear systems often evade traditional monitoring mechanisms due largely because minor physiological fluctuations frequently go unnoticed; notably sepsis carries a death rate between twenty to fifty percent while

over fifty million individuals suffer from epilepsy—with up to thirty percent being drug-resistant cases. Additionally rising antimicrobial resistance alongside unpredictable seizure patterns necessitates development pertaining towards predictive frameworks capable enough not only detect anomalies but also operate below conventional resolution thresholds afforded by standard tools. As of 2025, the use of complexity science to Intensive Care Medicine (ICM) remains poorly researched, with 0–3 articles published annually, which is a field that is well-versed in modeling emergent phenomena and nonlinear dynamics. Many PubMed papers relate these principles to the critical care. This gap highlights the need to create intelligent approaches using complexity science, advanced modeling (agent-based, system dynamics, hybrid), machine learning (ML), and large language model (LLM)-based multi-agent systems. These tools promise to transform how decisions are made in the ICU, allowing for early detection of abnormalities and ensuring that each patient receives the care that they need. Such breakthroughs will help with the global healthcare burden (e.g., sepsis costs more than \$20B a year in the U.S.). The Republic of Moldova, with its resource-constrained healthcare system, stands to benefit significantly from such innovations, thereby enhancing clinical precision and research capacity. This research plays a crucial role in connecting theoretical advancements to practical ICU applications, thereby aligning with global trends towards precision medicine and data-driven healthcare.

The aim of the study is to develop new intelligent methodologies, algorithms and software applications for acute care medicine, based on the complexity science paradigm. It aims to improve early detection, prediction and management of life-threatening illnesses, such as sepsis and epileptic seizures in the severely unwell. This research brings together ideas from nonlinearity, emergence and adaptation with advanced modeling techniques (agent-based, system dynamics and hybrid), machine learning, representation learning, estimation of algorithmic complexity, and multi-agent systems based on large language models in an effort to connect theoretical insights from the field of complexity science to practical tools for better clinical decision-making, personalized care and resource optimization in intensive care units.

Research objectives:

1. A complexity science approach: Develop a conceptual framework based on complexity scientific principles to study intensive care systems, separate it from systems theory, and apply it to model critical situations in sepsis and epilepsy.
2. Assessment of modeling techniques: Study of agent-based modeling for emergent behavioral simulation in physiological processes, such as cardiogenic pulmonary edema, in an effort to assist decision making in intensive care.

3. System dynamics modeling exploration: Explore system dynamics methods for modeling aggregate circulatory and feedback dynamics, including myocardial infarction scenarios, to enable efficient simulations in critical care.

4. Hybrid modeling methods development: Develop hybrid models that combine agent-based and system dynamics modeling into hybrid frameworks for multi-scale simulations, e.g. maternal-placental-fetal oxygen supply, to solve gaps in obstetric intensive care monitoring.

5. Large datasets identification and analysis: Identify and execute exploratory analysis on huge datasets for sepsis prediction and epilepsy research comprising EEG recordings and physiological time series to enable the development of standard artificial intelligence (AI) systems.

6. Data processing and imputation: Develop improved imputation, transformation and preprocessing techniques, including assessment of algorithmic complexity by block decomposition, for missing value handling and data preparation in machine learning applications.

7. Representation learning integration: Employ representation learning methods (e.g., NLP models like Word2Vec and BERT) to turn physiological data (e.g., EEG signals) into actionable representations for enhanced prediction accuracy.

8. Machine learning models investigation: Investigate various machine learning algorithms, enriched with algorithmic complexity measures, to improve diagnostic and prognostic capabilities in sepsis and epileptic seizure prediction.

9. Integration of large language models: Examine large language models with retrieval-augmented generation and multi-agent systems for improving sepsis management, guideline adherence, and clinical decision-making in intensive care.

10. Software development: Develop integrated software for early sepsis prediction, EEG analysis, and multi-agent sepsis management verified for clinical and research environments to support individualized therapies

Research hypothesis: Subtle pathophysiological changes that are below the resolution level of standard monitoring methods are often missed in the intensive care, delaying early identification and prediction of critical illnesses such as sepsis and epilepsy. Based on complexity science (nonlinearity, emergence, adaptation, and algorithmic complexity (e.g. Kolmogorov-Chaitin via block decomposition)), we hypothesize that the integration of advanced modeling (agent-based, system dynamics, hybrid), data imputation, representation learning (e.g. NLP on physiological signals), machine learning and LLM-based multi-agent systems will enable the detection of these changes, resulting in prediction systems with a performance that meets or exceeds literature benchmarks (e.g. AUC >0.95 for sepsis). The resulting software tools for EEG

analysis (research) and sepsis management (clinical) will assist in decision-making, individualized care and outcomes in Moldovan healthcare settings.

Synthesis of Methodology and Rationale of Chosen Methods. The project adopts a transdisciplinary approach grounded in complexity science, combining computational modeling, data pre-processing, ML and LLM-based multi-agent systems. Agent-based modeling (ABM) and system dynamics modeling (SDM) can be used to mimic the emergent and aggregate dynamics of physiological processes, including nonlinearities and multiscale interactions in ICU settings (e.g., pulmonary edema, fetal oxygen delivery). The hybrid ABM-SDM models are free from the limits of the standalone techniques, and enable complete simulation. For example, preprocessing of data like KCC with block decomposition and imputation (e.g. Kalman-ML) can make heterogeneous biomedical data suitable for ML by handling missing values and detecting patterns in non-stationary signals. We select ML techniques (e.g., GBM, Word2Vec) and NLP-based representation learning due to their high performance in forecasting tasks. LLM multi-agent systems with RAG improve decision-making by integrating knowledge in real time, tackling the complexity of sepsis management. These strategies are consistent with the thesis's objective of creating intelligent tools: the emphasis on emergence and adaptation in complexity science offers novel solutions for the ICU.

Summary of the Chapters of the Thesis. The thesis is organized in eight chapters, each one building on the previous, to study systematically intelligent methods and algorithms for intensive care from a complexity science perspective. This sequence follows a logical path from theoretical grounds to practical implementation and is highly relevant to the research aim of developing novel tools for the early diagnosis, prediction and management of critical conditions such as sepsis and epilepsy and the 10 research objectives.

The thesis comprises chapters, an introduction, conclusions and recommendations. The work is composed of 290 pages, 38 tables, 87 figures and bibliography of 387 titles.

Chapter 1: Conceptual Framework—A Complexity Science Approach to Intelligent Methods in Intensive Care

This chapter lays the foundation by covering the principles, history and application of complexity science in intensive care medicine. It introduces the main ideas of nonlinearity, emergence, self-organization, adaptation, feedback loops, networks and universality as well as comparing complexity science with systems theory to emphasize their differences and synergies. A PubMed analysis was performed to analyze the publication trends and relevance to intensive care, and to justify the choice of sepsis and epilepsy as representative complex adaptive systems. The paper also discusses the relationship between complexity science, modeling paradigms (e.g.,

agent-based and system dynamics modeling) and machine learning and proposes a new notion of complexity data science. The importance of this chapter is to set the conceptual framework (Objective 1) as it gives a theoretical scaffold that guides all subsequent inquiries and ensures that the research is based on an interdisciplinary approach that can address the complex dynamics of ICU environments.

Chapter 2: Agent-Based Modeling in Intensive Care: A Complexity Science Approach

Here, the focus is on the assessment of agent-based modeling (ABM) as a tool based on complexity science to simulate emergent behaviors of critical illness. Using a case study of cardiogenic pulmonary edema (CPE), the chapter presents the mathematical formalism of ATM, including agent-state updates, interaction probabilities, and quantification of emergent properties. The model is implemented in NetLogo and experiments are performed to verify the sensitivity of parameters and extend it for clinical decision support and education. This investigation is important for Objective 2, because the ability of ATM to model heterogeneous agents and spatial dynamics allows the capture of unpredictable micro-level interactions in physiological processes that are needed to develop intelligent methods that improve ICU triage and intervention planning, directly contributing to the overall aim of innovative simulation tools.

Chapter 3: System Dynamics Modeling in Complexity Science: Applications to Intensive Care Medicine with a Focus on the Human Circulatory System

This chapter discusses system dynamics modeling (SDM) as a complement to ATM in dealing with aggregate, continuous processes. It considers SDM formalism through ordinary differential equations (ODEs), state-space representations and numerical integration techniques, with a particular implementation in the circulatory system in the case of myocardial infarction. The exploration includes integration with lumped parameter models, digital twins and real-time data assimilation. This chapter is an essential part of Objective 3 and addresses the need for computationally efficient models that can deliver causal conclusions regarding feedback loops and nonlinearities, enabling macro-level simulations that are vital for proactive care in ICUs and for closing gaps in understanding systemic behaviors in critical conditions.

Chapter 4: Hybrid Modeling Approaches in Intensive Care Medicine: Integrating System Dynamics and Agent-Based Modeling for Enhanced Physiological Simulations

The present chapter builds on the previous ones by describing hybrid modeling, which couples SDM and ABM for modeling multiscale phenomena. This work is focused on the model for the simulation of maternal-placental-fetal interactions, especially fetal oxygen delivery, with macro-flows modeled by ODEs and micro-interactions by agents in NetLogo. The chapter covers synchronization challenges, simulations based on scenarios (e.g., anemia, abruptio placentae), and

extensions to digital twins. This is particularly relevant for Objective 4, as hybrid methods overcome limitations of each individual method, such as SDM's failure to account for heterogeneity or ABM's computational demands, providing comprehensive tools to forecast distress in complex settings (e.g., obstetric intensive care). This supports the objective by improving simulation precision and relevance.

Chapter 5: From Traditional Medicine to Computational Paradigms: Complexity-Driven Approaches to Diagnosis and Prognosis in Critical Conditions

This chapter discusses the historical evolution of medical diagnosis and prognosis from empirical methods to AI-based systems. It discusses mathematical and logical aspects (e.g., Bayesian, fuzzy logic, model-based), the relationship between diagnosis and prognosis, and specific applications in epilepsy (e.g., EEG analysis methods) and sepsis (e.g., evolving definitions and screening tools). We discuss representation learning and ML techniques at length. This study supports Objectives 7 and 8 by providing the rationale for transitioning to data-driven approaches to utilize the complexity science to find patterns in multivariate time series that are important to develop predictive systems that can detect subtle changes and customize care in intensive care units.

Chapter 6: From Raw Data to Actionable Insights: Imputation, Complexity Metrics, and Learning Representations for Critical Care Applications

This chapter is about data preparation. Imputation methods such as LOCF, custom interpolation and Kalman filtering, algorithmic complexity estimation using Kolmogorov-Chaitin complexity (KCC) with block decomposition and representation learning using TF-IDF, Word2Vec and BERT to transform EEG and sepsis data are considered. It analyzes epilepsy and sepsis datasets, such as exploratory visualizations and topological embeddings. These investigations are crucial for Objectives 5, 6, and 7 to ensure that raw biomedical data are converted into ML-ready formats, to capture nuanced patterns in non-stationary signals, and to address missing values, which are foundational for accurate prediction models and align with the aim's emphasis on intelligent algorithms.

Chapter 7: Integrating Algorithmic Complexity in ML for Intensive Care Applications: Findings, Discussions, and Practical Implications

In this chapter we build up on previous work and consider the use of ML models enriched with KCC for prediction of sepsis and analysis of EEG signals. It follows common steps in ML development, splitting data, selecting a model (e.g., GBM, DRF or DNN) and methods to make models easier to understand (e.g., SHAP or partial dependence plots). The practical use of software prototypes such as ISAAC-EEG-BDM-NLP and ISAAC Sepsis Expert is explored. This supports

Objectives 8 and 10, by emphasizing that integration of complexity metrics into ML workflows is necessary to successfully improve diagnostic performance and create deployable tools, which directly support the research purpose of enabling early detection and decision support in both clinical and research environments.

Chapter 8: Large Language Models and Multi-Agent Systems in Intensive Care: Enhancing Sepsis Management

The last chapter discusses large language models (LLMs) with retrieval-augmented generation (RAG) and multi-agent systems for sepsis management. It covers LLM architectures, RAG for knowledge integration, multi-agent collaboration via Markov processes, and case studies mimicking ICU workflows. This sub-section is essential to Objective 9, to address the need for advanced AI to support dynamic, evidence-based decision making, reduce hallucinations, and improve adherence to guidelines, and thereby achieve the aim through novel systems that improve precision medicine in the intensive care setting.

These chapters follow a logical sequence: from theory (Chapter 1) to modeling (2-4), data handling and ML (5-7), and AI applications (7 and 8). Each investigation is chosen with care to progressively reach the objectives, ensuring the development of comprehensive, complexity-informed tools for intensive care.

The thesis was developed in the Department of Software Engineering and Automatics at the Technical University of Moldova, as well as through various research projects:

- Postdoctoral project no. 25.00208.5007.07/PD "Intelligent Software Methods and Algorithms for Intensive Care: A Complexity Science Approach," funded by the National Agency for Research and Development (NARD) of Moldova;
- Project PN-IV-PCB-RO-MD-2024-036 "DACISLAB: Virtual laboratory on open data and open science in the new generation of computing continuum systems";
- International private research project "Algorithmic Information Dynamics," subproject "AID & (Patho)physiological and Clinical-end Models" (Appendix 1).

Publications on the topic of the thesis: The research resulted in 54 scientific works, with 20 as the sole author, 34 as the first/last author, including 25 articles of which 14 publications in journals/editions with an impact factor.

1. CONCEPTUAL FRAMEWORK - A COMPLEXITY SCIENCE APPROACH TO INTELLIGENT METHODS IN INTENSIVE CARE

1.1. Background

Complexity science (CS) is a new interdisciplinary field that aims to understand how many interacting components behave as systems [1, 2, 3, 4, 5]. Instead of reductionism (breaking phenomena down into simpler parts) in traditional scientific paradigms, CS focuses on the holistic nature of systems where the whole has properties which cannot be explained by analysis of its individual elements only. This is important for intensive care medicine because patient conditions are dynamic interactions among physiological processes, medical interventions, environment, and even social elements of healthcare teams.

Figure 1.1 shows results of a PubMed search with “complexity science” term (blue color bars) and “intensive care” (orange color bars) in title/abstracts for the years 2000-2025.

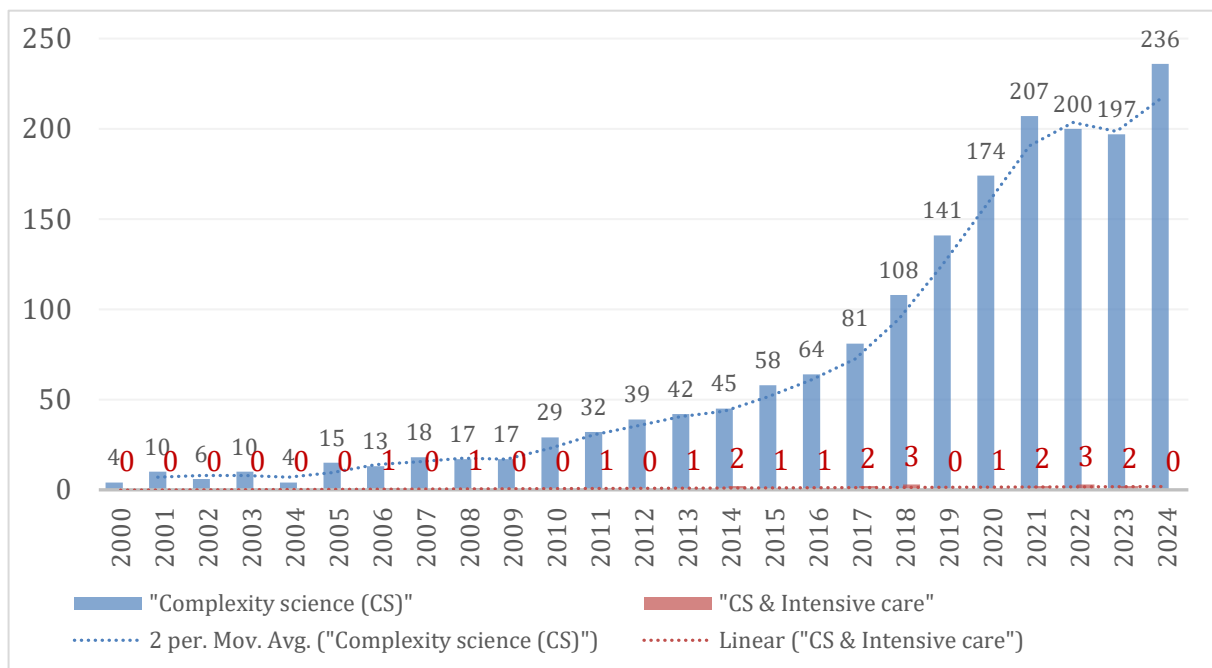


Figure 1.1. The number of publications in PubMed: blue – complexity science (exact match) and orange – in association with intensive care (a collection of terms like “intensive care”, “critical care”, “critical state”, “critically ill” grouped by OR operator).

One can see an increasing trend in publications on CS. But the association between “CS and intensive care” has a flat trend (publication numbers are 0–3 per year).

This background section describes where CS came from, how it changed over time, its main ideas, methods and how they are used; compares CS to systems theory (ST) to point out their differences; discusses specifically what relevance the CS has to healthcare and how it frames the research presented here on intelligent methods, and algorithms for intensive care.

1.1.1. Historical Development of Complexity Science

The history of CS goes back to middle of 20th century but its ideas go much further back to philosophy and early scientific research. Henri Poincaré's work on nonlinear dynamics and the three-body problem in the late 1800s showed sensitivity of initial conditions (a central idea in chaos theory). But formal foundations of CS did not emerge until the 1940s/50s with cybernetics and information theory. Norbert Wiener's cybernetics [6] described feedback loops (previously described by others) as mechanisms for control and communication in animals and machines; Claude Shannon's information theory gave a mathematical basis for quantifying uncertainty through entropy [7].

The 1960s saw chaos theory come along - Edward Lorenz's weather models showed the butterfly effect (tiny changes in initial conditions can produce vastly different outcomes; deterministic predictability is broken) [8]. Biologists like Ludwig von Bertalanffy were also contributing at this time [9]. His general systems theory influenced early CS because it promoted open systems which exchange energy and matter with their surroundings. By the 1970s evolutionary ideas had entered CS because of people like Ilya Prigogine, who studied dissipative structures (systems that keep order by losing energy) and won the Nobel Prize in Chemistry in 1977 for work on nonequilibrium thermodynamics [10].

A turning point in the institutionalization of CS was the founding of the Santa Fe Institute (SFI) in 1984 by scientists such as George Cowan, Murray Gell-Mann, and Philip Anderson [2, 3, 4, 11, 12] - SFI became a center for interdisciplinary research bringing physicists together with biologists, economists and computer scientists. Stuart Kauffman's studies of self-organization in biological networks [13] and John Holland's genetic algorithms, which simulated evolution computationally [14], were major milestones. In the 1980s and 1990s network theory emerged (Albert-László Barabási on scale-free networks [15]), which showed that many real-world systems have "hubs" that make them robust but vulnerable to targeted disruption.

As we move into the 21st century, CS has grown rapidly: in the 2000s big data analytics came in; social media, genomics, and climate models became available for study; applications to the social sciences increased in the 2010s (Duncan Watts and Steven Strogatz' small-world network model [16] explained phenomena such as six degrees of separation); in the 2020s CS started interacting with AI and ML (pandemic modeling during COVID-19 and adaptation to climate change [17,18]). In 2025 there are recent advances like the SFI symposium on emergence across the sciences, debates about questions such as the origins of life and social behavior, new publications on foundational papers in CS. Satellite conferences like the Conference on Complex

Systems 2025 (CCS'25) have satellites on networks, computational social science, and biological systems to show what is happening now.

The "2021 Map of the Complexity Sciences" by Brian Castellani and Lasse Gerrits (updated implicitly through discourse) tracks this evolution visually [19, 20]. It maps timelines from Wiener's cybernetics in the 1940s to big data and AI integration in the 2020s (e.g., graph theory to network science branches to evolutionary biology to complex adaptive systems (CAS)). Trans-disciplinarity is shown by how fields can influence each other (Benoit Mandelbrot's fractals come from mathematics; Niklas Luhmann's social systems come from social sciences).

ST came from general systems theory (GST) between the 1920s and 1950s because Bertalanffy wanted to unify different scientific disciplines against reductionism. ST focuses on wholeness, hierarchies, and equilibrium, whereas CS builds on that by studying adaptation and emergence in nonequilibrium states (more recent literature says ST is all systems, but CS specifically CAS with nonlinearity [21, 22]). For example, ST might model a system as a closed loop for stability, but CS looks at how perturbations lead to new states.

1.1.2. Key Concepts in Complexity Science

At the heart of CS are concepts which describe unpredictability but patterns in complex systems: Nonlinearity: outputs are not proportional to inputs; small changes can cascade into large effects (e.g., tipping points in ecosystems or financial markets) [1, 22]; Emergence: properties emerge from interactions beyond individual components (e.g., flocking in birds or consciousness from neurons); Self-organization: spontaneous order without central control driven by local rules (e.g., ant colonies or crystals); Adaptation: CAS learn and evolve in response to environments (often at the "edge of chaos" - transitional zone between stability and flexibility). Feedback loops amplify (positive) or stabilize (negative) changes, while networks model interconnections showing robustness and vulnerability; universality shows similar patterns at different scales (fractals in lungs, power laws in city sizes).

In healthcare these concepts present diseases as emergent phenomena [23, 24, 25, 26] (sepsis is nonlinear inflammatory cascades; epidemics self-organize through social networks). New 2025 insights are CS in AI governance (complex models exhibit emergent behaviors like biological systems do).

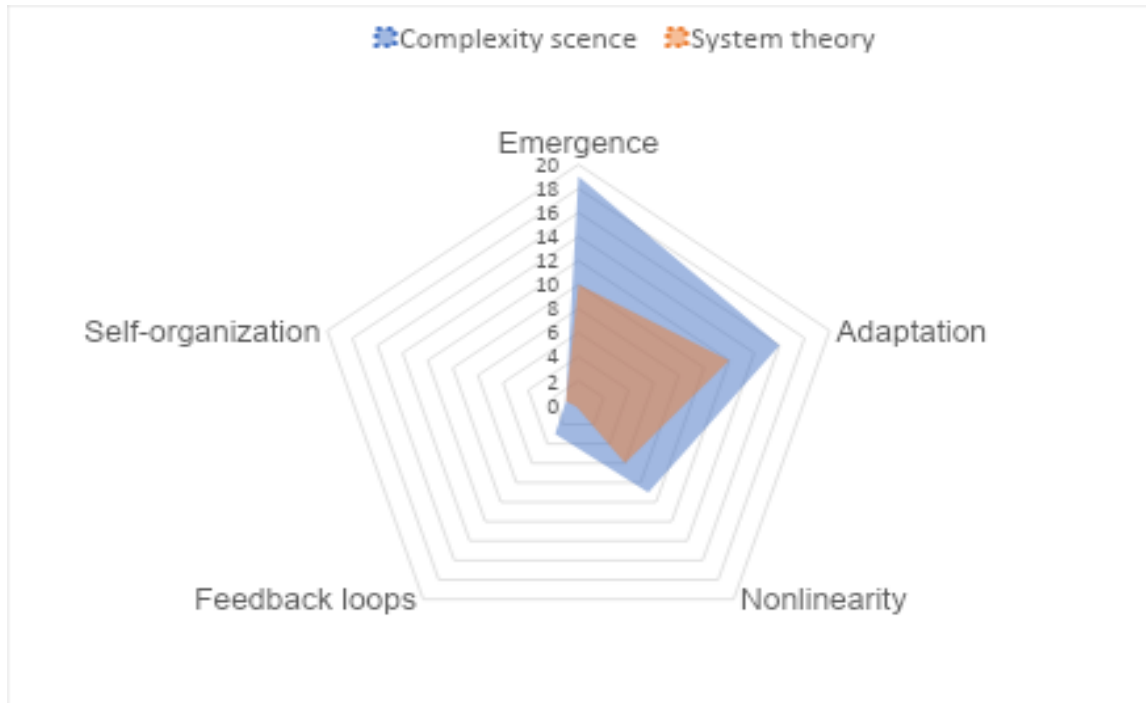


Figure 1.2. The weight of key concepts in percent for PubMed searches: blue – complexity science, orange – system theory (exact matches for the respective terms). Key concepts (e.g., emergence, adaptation, nonlinearity etc.) were added during the search as simple terms.

The search period is from 2000 to 2025.

1.1.3. Distinctions with Systems Theory

CS and ST have an interdisciplinary ethos but focus differently [27]. Historically GST (general systems theory) arose in reaction against reductionism in the mid-20th century (homeostasis hierarchies). CS only emerged later and combines chaos (dynamics) with computation [21, 28].

ST focuses on structural stability through feedback and boundaries; CS emphasizes nonlinearity, emergence, and self-organization in evolving systems [27]; methodologically ST uses qualitative and mathematical models while CS simulations (ABM, etc.) [26, 29, 30, 31] - applications: ST for control in engineering; CS in ecology for adaptation. Similarities are the rejection of reductionism, focus on emergence and interdisciplinarity. Together they are complementary tools for complex problems.

Established literature treats CS as a special case extension or “offshoot” of ST rather than a replacement (Richardson [32] says that ST is a philosophy against reductionism, but CS introduces a “problematization of existence” through nonlinearity and unpredictability, so computational tools like ABM are needed to model phenomena which traditional ST models might treat qualitatively or statically). Ladyman et al. [33] do the same thing. Complex systems in CS

terms are those which emerge and self-organize beyond simple hierarchies (CS differs from earlier ST frameworks where stability and boundaries were emphasized [11]).

Table 1.1 shows how anti-reductionist ethos, interdisciplinarity, and wholeness complement each other rather than oppose one another.

Table 1.1. System Theory & Complexity science: similarities

Aspects	Systems Theory & Complexity Science
Holistic Perspective	Both system theory and complexity science view systems as wholes not parts - reductionism is opposed to holistic thinking since it breaks systems down into smaller pieces. When we understand interdependencies and interactions between components, both paradigms try to capture emergent properties.
Interdisciplinary Nature	ST/CS are inherently interdisciplinary concepts and methods drawn from biology, physics, sociology, economics, and engineering. This interdisciplinary nature allows for the application of these paradigms across a broad range of disciplines so that complex phenomena can be explored in different contexts.
Focus on Emergence	One similarity is that both systems theory and complexity science focus on emergence (new properties or behaviors emerge in the system but cannot be predicted by looking at the parts of the system separately). System Theory and Complexity Science, both want to understand how emergent properties manifest and influence overall behavior.
General Characteristics	Systems theory and complexity science are two paradigms for understanding complex systems: Systems theory provides a structured, holistic way of analyzing organized systems (stability and feedback); complexity science, with its focus on adaptation, emergence, and nonlinearity, explains what makes complex adaptive systems dynamic and evolving. Together they add to our understanding of complexity tools and perspectives for many scientific and practical applications. By combining strengths of both paradigms, researchers and practitioners can develop more comprehensive strategies to manage and understand the intricate systems that characterize our world.

1.1.4. Applications of Complexity Science in General and in Healthcare

CS applies to physics (disordered systems), economics (market crashes), and biology (ecosystems) [33]. In healthcare it models systems as CAS to deal with nonlinearity in patient responses. Scoping reviews show CS in health services research [23, 24] using concepts like self-organization to implement evidence; in 2025 CS informs AI in healthcare (diagnostics and policy) [24, 25].

In intensive care, CS sees sepsis as a chaotic inflammatory state or transition to organ failure [26]. Implementation science uses CS to navigate dynamic interventions [25].

CS underpins this thesis by seeing intensive care as CAS, which informs models for sepsis and epilepsy; guides preprocessing with Kolmogorov-Chaitin complexity [34,35], based on

Solomonoff [36], Kolmogorov [37], and Chaitin [38]; ML for prognosis [39, 40]. Thus, CS enables multidisciplinary innovation.

1.2. Complexity Science Interactions with Agent-Based Modeling (ABM), System Dynamics Modeling (SDM), and Hybrid Approaches

CS interfaces with modeling to simulate CAS dynamics. In this section ABM, SDM, and hybrids are described (with healthcare examples).

Agent-Based Modeling

ABM models autonomous agents interacting according to rules so emergence is captured [29, 41]. In intensive care (IC/ICU), ABM models COVID-19 ICUs, or antimicrobial de-escalation. Thesis Chapter 2 uses ABM for cardiogenic pulmonary edema.

System Dynamics Modeling

SDM employs stocks, flows and loops for aggregate dynamics [18, 26, 30]. In healthcare SDM plans interventions for cardiovascular disease (CVD) or workforce management. SDM are used in Chapter 3 of the thesis to model circulatory systems. In healthcare, SDM is used to plan interventions for cardiovascular disease (CVD) or to manage the workforce. Thesis Chapter 3 uses SDM for circulatory systems.

Hybrid Modeling (SDM & ABM)

Hybrids combine macro and micro views. Medicine usually simulates health systems [26]. Thesis Chapter 4 models fetal oxygen delivery using this approach.

The interactions of CS with modeling are discussed in the thesis' chapters: ABM models micro-level agent interactions leading to macro-emergence (e.g., heterogeneous patient responses in ICUs); SDM aggregates flows and feedback loops to model system stability, and change; hybrid combines these for multiscale analysis (e.g., fetal monitoring where direct measurements are not possible).

Overall, the thesis' respective chapters show how CS can be used to model complex physiological disruptions (pulmonary edema from cardiac-pulmonary interaction, myocardial infarction as feedback-loop breakdown in circulatory systems, and indirect estimation of fetal oxygen delivery via adaptive models).

1.3. The Interplay of Complexity Science and Machine Learning: A Common View

We can think of CS and ML as mutually reinforcing and bidirectional. CS provides philosophical perspectives on emergent properties of ML models (neural networks acting like

complex adaptive systems), while ML provides instruments to model and analyze complex systems (hidden patterns in high-dimensional data [42, 43]).

Unifying complexity science and ML does not only increase predictive power but also provides new paradigms for tackling real-world problems. The heart of unification is that both areas struggle with the difficulty of finding regularities from irreducibly high-dimensional phenomena: Complexity science works on coarse-graining (breaking systems into simpler parts to find new or stable patterns), while machine learning is good at encoding dependencies by processing lots of data [42].

So, complexity science and machine learning are fertile ground for innovation since theoretical perspectives combine with practical instruments to tackle high-dimensional problems [42, 44].

Computer science and machine learning come together in the thesis to deal with nonlinearity; Kolmogorov-Chaitin complexity measures how much information needs to be processed on a dataset; applications in ML include randomness measures; Chapters 5 to 7 use these techniques for diagnosis.

1.4. Argumentation for Sepsis and Epilepsy as Central Pathological States in Intensive Care

The reason sepsis and epilepsy are selected as central pathological states in this thesis is because they are common in ICUs, in line with principles of complexity science and practically relevant to develop intelligent methods and algorithms:

a. Heavy Burden on Healthcare Systems

Sepsis and epilepsy cause major clinical, economic, and societal burdens, so they are priorities for innovation in intensive care. Sepsis is the leading cause of ICU admissions and deaths worldwide (average hospital stay cost for a patient in high-income country is more than US\$32 000), over 50% of hospital deaths; mortality rates range from 10–20% in uncomplicated cases to 46% in severe ones [45, 46].

Epilepsy - especially in its severe form as status epilepticus (SE) is a major emergency in the ICUs with substantial morbidity and mortality. SE has economic costs (long ICU stays and intensive management are more expensive than for other neurological conditions [47]).

So, complexity-based approaches might save money and improve outcomes; hence, they were chosen as examples of intelligent ICU interventions.

b. Examples of Critical States in Complexity Science

Sepsis and epilepsy are “critical states” - a central concept in complexity science - which is why these diseases are ideal models for nonlinear dynamics, self-organization and emergent behavior. In complexity science a critical state is the poised condition of a system at the edge of phase transition where small perturbations lead to large-scale changes (power law distributions, self-similarity, long-range correlations e.g., self-organized criticality), unlike stable (ordered) or chaotic (disordered) regimes, where the system can be adaptable and information processing efficient [24].

In medicine critical states emerge as essential points of physiological systems [48] and resemble the concept of critical state in anesthesia and intensive care [49].

For sepsis, this is seen in the rapid progression from local infection to systemic inflammation and multi-organ failure, where the immune response hovers around a critical point (nonlinear dynamics and emergent cascades, e.g., cytokine storms). Epilepsy (especially seizures or SE) shows neuronal networks near criticality (synchronized firing patterns scale free avalanches, phase transitions from normal brain activity to hyperexcitability, etc.). These properties match complexity science’s focus on emergence, so algorithms can be developed that detect early warning signs of transitions (fractal analysis/network theory etc.).

c. Large-Scale Datasets

The large number of high-quality publicly available datasets for sepsis and epilepsy allow testing and validation of complexity-based methods, which strengthens the thesis’ empirical basis. For sepsis, PhysioNet/Computing in Cardiology Challenge 2019 has over 40 000 patient cases from ICU electronic health records with vital signs, lab values, and demographics for early prediction tasks [50].

A large set of data comes from the American Epilepsy Society (AES). This set offers more than 8,000 EEG clips of intracranial recordings to analyze seizure [51]. Multimodal datasets linking sepsis to post-seizure risk (e.g., stroke survivors) provide thousands of cases for integration studies.

These large-scale repositories allow scalable algorithm development that is lacking in other ICU pathologies.

d. Suitability for Kolmogorov-Chaitin Complexity Estimation via Block Decomposition Method

The multivariate multimodal time series nature of sepsis and epilepsy data makes them particularly well suited for estimation of Kolmogorov-Chaitin complexity (KCC) using the Block Decomposition Method (BDM), a key transformation technique in this thesis for data analysis and machine learning. KCC measures the shortest program required to describe a dataset; it quantifies

randomness and structure inherent to non-computable systems (i.e., signals that are irregular). In sepsis datasets, heart rate variability, blood pressure, inflammatory markers, etc. are high-dimensional time series which BDM can decompose into blocks for approximation of complexity and patterns for predictive modelling (early deterioration, etc.) emerge. For epilepsy, EEG and multimodal signals (e.g., EEG with vital signs) enable BDM to estimate complexity in seizure dynamics that can be used for feature extraction by ML classifiers. Raw data are converted into interpretable complexity metrics for anomaly detection and personalized interventions in the ICU [52].

Thesis chapters on ABM and SDM modeling approaches are also critical state related (Chapter 2 pulmonary edema as CAS; Chapter 3 myocardial infarction simulation; Chapter 4 fetal oxygen estimation via models focusing on abruptio placentae). In these chapters KCC is also central.

1.5. Complexity Data Science: Reframing Intelligent Methods in Intensive Care

Complexity data science (CDS) is a new interdisciplinary field combining basic ideas of CS with analytical tools from DS [5]. It is an elegant way to look at the intelligent methods and algorithms developed in this thesis for intensive care. CDS was coined as a “spin-off” of the digital twin (DT) concept - virtual replicas of physical systems used for simulation, prediction, and optimization — because high dimensional dynamic data in CAS cannot be handled with traditional DS. Traditional DS views data as samples from statistical distributions; CDS views generative processes that produce data instead (CS ideas such as self-organization, emergence, and nonlinearity). This is particularly well suited to intensive care, where patient data streams from physiological monitors, EHRs, and interventions form complex time-varying networks that are not amenable to reductionist analysis.

CDS is based on the historical development of CS discussed earlier but extended to the data-driven 2020s. CDS started with DT technology (applied initially in engineering and manufacturing for real-time simulations of physical assets) [5]. Scientometric studies show that DT applications have spread across healthcare (personalized modeling of patient trajectories) [5]. CDS does not replicate but extracts emergent patterns from messy data (big data analytics, ML) to reveal hidden interactions. This is where CS institutional milestones (e.g., Santa Fe Institute trans-disciplinarity) are now amplified by data-intensive methods; big data in the 2000s and AI in the 2020s, as discussed in historical development, fit naturally into CDS, where DT-inspired frameworks study phenomena like pandemic modeling (similar to ICU environments [17]).

CDS gives key CS concepts operational value in data-rich settings. Nonlinearity, emergence, and self-organization are no longer abstract; they become measurable through data decomposition techniques that expose scale-free patterns and phase transitions in datasets (universality - patterns repeating across scales can be detected via statistical analysis and network modeling in CDS; PubMed trends (Figure 1.1) turn into actionable insights). CS has a flat relationship with intensive care publications (0–3 per year), but CDS could fill this gap by using visualization and exploratory analysis to find hidden connections in clinical data [5]; feedback loops and adaptation at the “edge of chaos” are recast as learning mechanisms in CDS models where ICU monitors self-organize into predictive schemas. In healthcare, CDS views diseases as emerging from data-generating processes (nonlinear cascades in sepsis or scale-free neuronal avalanches in epilepsy), coarse-graining high-dimensional data so stable patterns emerge [23, 24].

Distinctions and synergies with systems theory (ST) sharpen under CDS. ST looks at structural hierarchies and equilibrium, whereas CDS looks at dynamic data flows in nonequilibrium states (DS tools such as uncertainty quantification are used to deal with perturbations). Similarities (anti-reductionism, interdisciplinarity, Table 1.1) are amplified in CDS education, where CS concepts (e.g., complex networks, phase transitions) are combined with DS methods (deep learning, natural language processing) to create a holistic curriculum (Figure 1.3).

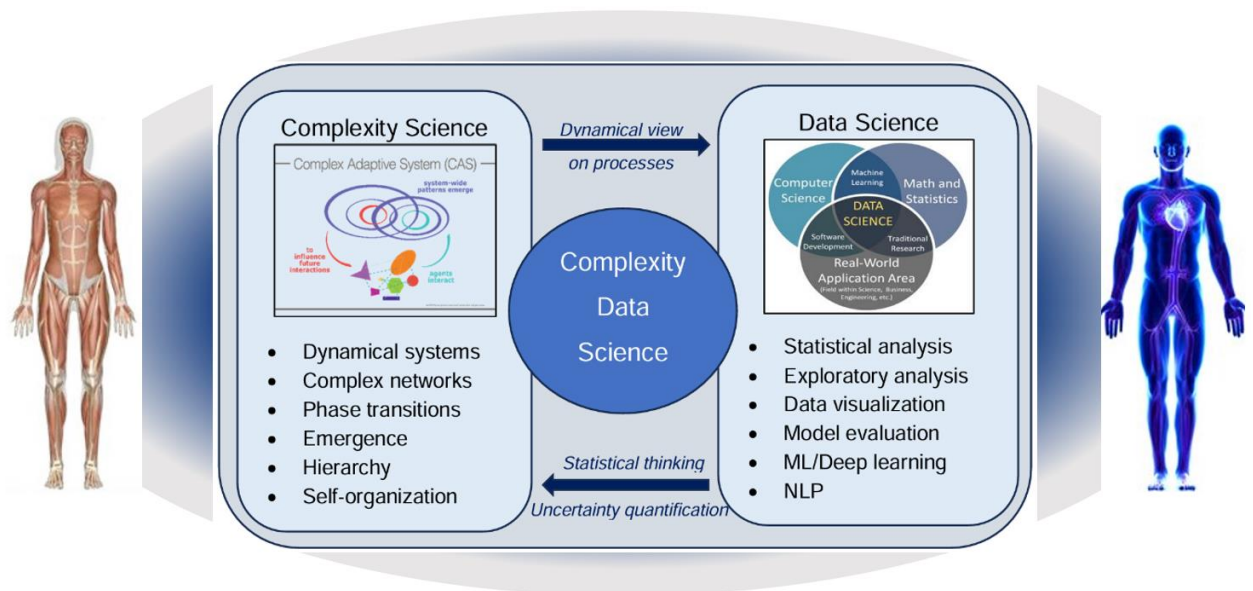


Figure 1.3. A holistic curriculum for CDS. Based on [5].

This unifying perspective fills educational gaps: DS doesn’t have a dynamic view of data generation; CS may not be statistically rigorous enough. CDS does this by promoting hybrid

training so researchers model ICU systems where data emerge from interacting agents (small world networks in social epidemiology).

The interplay between CS and modeling makes CDS coherent: In intensive care CDS sees models as digital copies of physiological systems. Data science methods (model evaluation and deep learning) can find weaknesses (where an organ fails).

Bidirectional synergies with ML are central to CDS and unify the intersections of the thesis. CS explains emergent properties in ML models (e.g., neural networks as CAS), while ML scans messy data for patterns [30]. CDS broadens the topic by acknowledging that both fields have the same goal of coarse-graining high-dimensional phenomena and that techniques like KCC estimation via the block decomposition method (BDM) can be used to quantify intrinsic randomness in time series (i.e., CS). CDS frames this as a DT offshoot: ML models become virtual representations of ICU situations, so better prediction of critical states is possible. For sepsis, CDS can apply deep learning on cytokine-storm data to find nonlinear escalations; for epilepsy, it can use network analysis on EEG clips to detect self-organized criticality and personalize interventions.

CDS also justifies sepsis and epilepsy as pathological exemplars by saying they have lots of data and are aligned with critical states and then extends this to modeling chapters where we see pulmonary edema, myocardial infarction and abruptio placentae as CAS generating data. CDS provides hybrid simulations that can be used to produce indirect estimates.

CDS also turns the conceptual framework of the chapter on its head (CS is data-centric evolution) and ties theory to practice in ICU (DT spins off from CS); intelligent methods get tools for dealing with complexity so innovations in modelling, ML, and clinical applications can emerge. This lens not only shows what ICUR does not do yet but puts the thesis at the forefront of transdisciplinary advances that will improve outcomes in dynamic healthcare systems.

Research Problem Statement and Methodological Approach

On the basis of the goal to develop intelligent methods and algorithms based on complexity science for intensive care, this research aims to increase early diagnosis, prediction, and management of sepsis and epilepsy and other critical illnesses; a complexity framework needs to be established; modeling techniques need to be evaluated; datasets analyzed; software tools created. To achieve these goals, agent-based/system dynamics/hybrid modelling simulate physiological dynamics; Kolmogorov-Chaitin complexity preps data; representation learning and machine learning (e.g., NLP, BERT) predict; large language model multi-agent systems support

sepsis management. Data are taken from public datasets so scalable, interpretable solutions can be provided for Moldova's healthcare.

1.6. Synthesis of Chapter 1

This chapter is the basis for the thesis, as it looks at innovative methods and algorithms in intensive care from a complexity science perspective: Complexity science is an interdisciplinary field that studies systems with many interacting components where wholes arise (behaviors) out of parts, which cannot be reduced to their individual parts (reductionism); this goes against traditional reductionist thinking and is very relevant to intensive care because outcomes depend on physiological processes, medical interventions, environment, and healthcare team dynamics (see below). It describes the origins, evolution, principles, and applications of complexity science (distinguishing it from systems theory) and how it shapes the research on intelligent tools for critical care in the thesis.

The PubMed publication counts show a steady increase in interest from 2000 to 2025, but complexity science's association with intensive care is still low (0-3 papers per year) (Figure 1.1); there is room for more integration into medical research.

Complexity science as an area of research dates back to late 19th-century ideas on nonlinear dynamics but formalizes in the mid-20th century with cybernetics and information theory (feedback loops, entropy measures); chaos theory in the 60s showed sensitivity to initial conditions (butterfly effect); evolutionary ideas in the 70s (dissipative structures in nonequilibrium systems). Key institutions emerged in the 80s that allowed interdisciplinary work on self-organization, genetic algorithms, and network theory (scale-free models in the 90s). In the 21st century, big data, social sciences, artificial intelligence, and real-world applications like pandemic modeling have been added. Today symposia on emergence are held, and conferences on networks and biological systems exist (as of 2025). Visual maps organize timelines from cybernetics to AI integration - trans-disciplinarity between fields (mathematics, social sciences, etc.).

Systems theory came later as a unifying framework against reductionism (wholeness hierarchies equilibrium). Complexity science builds on this, but emphasizes adaptation and emergence in nonequilibrium, nonlinear settings. Differences are that systems theory focuses on structural stability and qualitative models, whereas complexity science is dynamic simulations and computational tools; similarities are anti-reductionism, emergence, and interdisciplinarity (Table 1.1).

Core concepts in complexity science capture the unpredictable structured nature of complex systems: nonlinearity shows disproportionate effects from small inputs (tipping points), emergence

properties emerge from interactions (collective behaviors in biological or social groups); self-organization spontaneous order from local rules; adaptation at chaos' edge (stability/flexibility balance); feedback loops amplification/damping; networks interconnections vulnerabilities; universality repeated patterns across scales. In healthcare these frame diseases as emergent from interacting elements (nonlinear cascades in infections or self-organizing epidemics). A PubMed weight analysis (Figure 1.2) shows how much these concepts have been searched for in complexity science compared to systems theory over the past quarter century.

The applications go from physics to economics to biology, but in healthcare, complexity science models systems as complex adaptive entities - nonlinearity of patient response and evidence implementation. In intensive care it looks at chaotic inflammation in sepsis or transitions in organ failure. The thesis uses this knowledge to see intensive care as a complex adaptive system (models for sepsis and epilepsy), preprocesses data with complexity measures, and applies machine learning for prognosis.

The chapter then goes into more detail about how complexity science and modeling techniques work together. Agent-based modelling simulates individual agents interacting to reveal emergent macro-behaviors (e.g., cardiogenic pulmonary edema in intensive care units) used in the thesis; System dynamics modelling stocks, flows, and loops aggregate behavior of systems (here circulatory systems in myocardial infarction); hybrid approaches merge these for multiscale views (e.g., fetal oxygen delivery where direct measurements are not possible). These methods fit complexity science's micro-macro dynamics feedback adaptation paradigm, allowing simulations of physiological disruption as emergent phenomena.

Interactions with machine learning are bidirectional and unifying. Complexity science explains emergent properties in neural networks as complex adaptive systems, and machine learning looks for patterns in high-dimensional data. Both fields are about extracting regularities from complexity (coarse-graining / dependency encoding). In the thesis this intersection handles nonlinearity via complexity quantification (diagnosis, large language models).

Sepsis and epilepsy were chosen as focal pathologies for several reasons: they are heavy burdens on healthcare (sepsis causes ICU mortality and cost; epilepsy, especially status epilepticus, is resource intensive); they are critical states in complexity science (poised at phase transitions with power law behavior and adaptability - sepsis escalates quickly or neuronal avalanches in epilepsy), datasets exist (thousands of cases public repositories), so testing can be done robustly; multivariate time series fit Kolmogorov Chaitin complexity estimation via block decomposition (data into metrics for anomaly detection and personalized care), etc. This extends

to modeling chapters in the thesis on related critical states (e.g., pulmonary edema, abruptio placentae).

Complexity data science is seen as a new paradigm that combines complexity science and data science (Figure 1.3) derived from digital twins (virtual system replicas used for simulation and optimization). Data are viewed as produced by dynamic processes, emergence, and nonlinearity in high dimensions, static data science not being quantified through decomposition and network analysis but healthcare data streams re-framed as evolving networks; historical/conceptual aspects of complexity science are amplified; gaps in intensive care research filled; different from systems theory because it looks at flows in nonequilibrium states; synergies between ideas help create hybrid education curricula (complexity ideas + data methods). In modeling simulations are digital twins that reveal vulnerabilities; with machine learning coarse-graining unifies sepsis and epilepsy critical-state predictions.

So, this conceptual framework positions complexity science as central to intelligent methodologies development in ICU (tools to manage nonlinearity, emergence, and adaptation); how to do the work? Future work should focus on hybrid models (agent-based and system dynamics approaches); multiscale simulations of ICU scenarios. Attention to Kolmogorov-Chaitin complexity (block decomposition) will help with data preprocessing for machine learning, which can detect sepsis and epilepsy early on. Data science should be used to tackle complexity in the thesis (digital twin-inspired models) so precise predictive algorithms can be developed that improve patient care through innovative teamwork in changing healthcare systems.

2. AGENT-BASED MODELING IN INTENSIVE CARE: A COMPLEXITY SCIENCE APPROACH

2.1. Agent-based modeling in medicine and complexity science

Complexity science (CS) is a way of seeing systems that have complex nonlinear relationships, emerging behaviors, adaptability, and self-organization. CS is not a reductionist technique that breaks down systems into discrete components; it is more of an "all the pieces fit together" kind of thing, so it is a suitable candidate for medical research, where physiological cellular and clinical systems are highly complex. CS provides a model for dealing with the unreliability of the emergence of patients with complex life-threatening diseases in an intensive care setting where fast and precise responses are required. ABM is a computational method that simulates interactions between autonomous agents to reproduce system-level behaviors. ABM is an effective tool for investigating intensive care scenarios. This section discusses an area where CS meets ABM in medicine that applies to intensive care.

2.1.1. Evolution of Complexity Science in Medicine

The science of complexity developed in the late 20th century from cross-disciplinary work at institutions like the Santa Fe Institute, where Murray Gell-Mann, John H. Holland, and Stuart Kauffman set down basic concepts for looking at CAS [53]. CAS are nonlinear (small changes in inputs lead to disproportionate results) and emergent (system-level features emerge from local interactions that cannot be predicted by studying components separately) [54]. These qualities appear in clinical systems (hospital operations or care pathways [55]) and medicine (immunological responses or cardiovascular dynamics).

CS is especially relevant in intensive care because critical states such as sepsis, acute respiratory distress syndrome (ARDS), or cardiogenic pulmonary edema (CPE) are complex situations with nonlinear connections between several physiological systems (cardiovascular, respiratory, and immunological) that may lead to emergent phenomena (organ failure/system instability) [56] (e.g., sepsis is characterized by a cascade of inflammatory mediators that increase/decrease responses differently, so it is nonlinear dynamics [57]). CS provides us with a window into that association and how interventions can stabilize or destabilize system functions. ST is for structure, but CS is for adaptation, self-organization, and the "boundary of chaos," where systems are most flexible and creative (stability and feedback loops maintain homeostasis, e.g., negative feedback in blood pressure regulation).

CS is interdisciplinary (biology, physics, math, and computer science), so it works well with many challenges in acute care: clinicians have to address logistical and organizational issues too

(resource planning, individual triage, etc.) [58]. CS's holistic strategy (not reductionist) allows scholars to see emergent qualities (e.g., the significant effect small changes in ventilator settings have on clinical results).

2.1.2. Agent-Based Modeling: A Computational Procedure

ABM simulates systems by having individual agents (cells, molecules, patients, or physicians) follow rules in a virtual environment. These contacts produce emerging behaviors [59], so ABM is well suited to represent the complexity of medical systems. ABM has heterogeneity and spatial dynamics plus stochasticity, which are essential in the ICU, where patient-centeredness and timing matter. Usually, mathematical models describe aggregate dynamics via differential equations [60]. ABM is medical in origin, coming from computational biology and first used for epidemiology and oncology (Epstein 2009). It simulated infectious disease transmission with ABM and found patterns undetectable by typical compartmental models [61]; An (2003) first applied ABM to cellular biology mimicking mechanisms important for pulmonary edema [62].

ABM's ability to simulate individual agents and their interactions makes it especially useful in acute care. For example, in CPE ABM may represent fluid flow where agents are water molecules controlled by hydrostatic and oncotic pressure. It can model pathological floods caused by tiny changes in parameters [63]. ABM can also model operational aspects of intensive care, such as patient flow, staff scheduling, or ventilator allocation besides pathophysiology. Agents may represent patients, clinicians, and resources with certain behaviors [64]. They can capture emergent phenomena (e.g., ICU capacity bottlenecks) that conventional techniques miss.

2.1.3. Intensive Care Applications

ABM use in the ICU includes physiological, clinical, and operational domains. In physiology modeling, ABM can be used by medical students to use interactive model simulations to simulate cellular or molecular interactions (e.g., immunological responses in sepsis or fluid shifts in CPE) and can reveal information about illness progression and therapeutic effect (Sepsis ABMs can, for example, model cytokine interactions to learn how inflammatory cascades lead to organ damage and help develop personalized therapies [57]). In clinical applications, ABMs can replicate patient responses to interventions (vasopressor administration, mechanical ventilation, etc.) by modelling physiological parameters as agents affected by drug effects or device settings.

ABMs could also be a better way of managing ICU resources. Patient triage models, e.g., can simulate patient, staff, and equipment interactions to explore ways of reducing wait times or distributing ventilators optimally. These models are particularly useful in crisis situations (e.g., pandemics) where scarcity of resources increases complexity. ABM also improves medical

education by giving trainees the opportunity to look at detailed scenarios through computer simulations, which also have a visual, hands-on component. For example, students can change parameters of an agent-based model of CPE and see which hydrostatic pressure variations cause alveolar flooding so they have a better understanding of pathophysiology [66].

2.1.4. Difficulties and Prospects

ABM in the ICU has promise but also problems. Scalability is limited by processing complexity (e.g., many agents or complex environments). Accuracy must be guaranteed by comparison with clinical data; intensive care data are usually noisy or incomplete, so calibration is difficult [67]. ABMs need to balance usability (i.e., simplicity) with realism (i.e., clinical significance); achieving this balance is hard when modeling multifactorial conditions such as multi-organ failure.

Opportunities for ABM in the ICU are significant. Advances in computational power and platforms like NetLogo make ABMs easy to use. This allows investigators to build and exchange models faster [68], or they can be combined with real-world data from electronic health records or wearable sensors to allow personalized predictions (e.g., predicting CPE progression based on individual patient pressures). ABM also serves an instructional purpose, as interactive training can connect academic knowledge with clinical intuition and enhance advanced decision-making among ICU physicians [69]. In critical care, CS principles can transform research and practice in the intensive care unit by providing a scalable method to understand and influence complex systems.

2.2. Mathematical Model of Agent-Based Modeling

ABM is a computer modeling approach that simulates complex systems by modeling individual agents and their interactions inside a given context. Classical models describe overall dynamics using differential equations, whereas ABM describes the emergence of events by local rules (which are usually probabilistic). This section describes the mathematical model of ABM, with a special focus on its application to medical systems (intensive care units), including the special focus on its application to medical systems (intensive care units), including main equations and explanations.

2.2.1. Core Components of ABM

A set of independent agents, governed by rules, represents systems and interacts with one another. Agents are embedded into a spatial environment (usually a grid or network), where their interactions yield system-level outcomes [53]. The following components can describe the mathematical model of ABM:

1. *Agent States*: Each agent i has a state vector

$$S_i = [x_i, y_i, v_i, \dots], \quad (2.1)$$

where x_i, y_i are position coordinates, v_i is a physiological property (e.g., pressure, concentration), and t is the time increment. In medical ABMs the agents could be water molecules or cells or patients and their states could be their position or health condition respectively.

2. *Interaction Norms*: Agents interact based on predefined rules, often probabilistic, that reflect real-world processes. These rules can be expressed as conditional statements or functions. For example, in a model of pulmonary edema, the probability of water molecules extravasating may be:

$$P_{extravasate} = g(HP - OP), \quad (2.2)$$

where g is a function (e.g., sigmoid) translating the difference between hydrostatic HP and oncotic pressure OP to a probability.

3. *Environment*: The environment is usually a discrete grid or continuous space. In a grid environment is described by coordinates and attributes (permeability, capacity, etc.). Environment affects behavior of agents and can change through their actions.

4. *Time Evolution*: Simulations run in discrete time steps (ticks) and update agent states iteratively. The state at time $t + 1$ is a function of the state at time t :

$$S_{i_{t+1}} = f(S_{i_t}, N_{i_t}, E_t, R), \quad (2.3)$$

where S_i is the state of the system, N_i is status of surrounding agents, E_t is state of environment, and R is the set of rules.

2.2.2. Mathematical Modeling

The dynamics of an ABM can be expressed in terms of a few basic equations as follows:

- *Agent State Update*: The state of an agent is updated according to its current state, neighbors, environment, and rules:

$$S_{i_{t+1}} = f(S_{i_t}, N_{i_{t+1}}, E_t, R), \quad (2.4)$$

For example, in a CPE model a water molecule's position may be updated according to blood flow direction and pressure-driven extravasation.

- *Emergent Behavior*: Properties of the system level (e.g., total extravascular water volume) emerge from interactions between agents:

$$V_{total_t} = \sum_{i \in W} v_{i_t}, \quad (2.5)$$

where W is the set of water molecule agents, and v_{i_t} is the volume contribution of agent i .

- *Environmental Dynamics*: The environment changes as a consequence of agent actions.

ACM water content in pulmonary edema model is updated as follows:

$$E_{ACM_{t+1}} = E_{ACM_t} + \sum_{i \in W} \Delta v_{i_t} - D_{lymph}, \quad (2.6)$$

where Δv_{i_t} is the water extravasated by the agent i , and D_{lymph} is the rate of lymph drainage.

- *Random components in ABM (agent-based modeling) usually include randomness to describe unpredictability*. For example, the movement of a water molecule may follow a random walk with probability modulated by $(HP - OP)$ through a stochastic function:

$$P_{move} = random(0,1) < g(HP - OP). \quad (2.7)$$

2.3. Application to Critical Care

ABM formalism is well suited to explain dynamic events like fluid shifts in CPE in critical care (see the simplified Starling equation 2.8 used in the CPE model):

$$Q = k(HP - OP). \quad (2.8)$$

Here, fluid flux is Q , permeability constant is k , and $(HP - OP)$ causes extravasation. This equation is at agent level, i.e., movement of each water molecule is driven by pressure differential. This can cause unanticipated effects (alveolar flooding). Flexibility of ABM allows for geographic heterogeneity (e.g., varying lung permeability), stochasticity (e.g., random molecular movement), which are needed for simulating patient specific responses in intensive care [69].

ANOVA statistics are widely used to validate ABMS and evaluate parameter sensitivity (in the CPE model variation of and analysis of outcomes (e.g., time to exceed 400 mL) match clinical observations [69]). Spatiotemporal activity in ABM makes it a tool for modelling complex intensive care scenarios from physiology to clinical decision-making.

2.4. Agent-Based Modeling of Cardiogenic Pulmonary Edema: Model Description

Here we present an ABM built in NetLogo that models CPE (life threatening illness where fluid builds up in lungs from increased HP and changes in OP). It shows CPE as a dynamic process dependent on HP and OP (with special focus on accumulation of extravascular lung water). It provides a comprehensive geographical context for understanding fluid mechanics, physiological balance, and disease development in critical care settings.

2.4.1. Model Structure

The ABM is implemented in a reduced physiological space divided into three zones:

1. Capillary: Space in blood vessels through which blood flows.
2. Alveolar-Capillary Membrane (ACM): ~400 mL border with fluid extravasation.
3. Alveoli: Air-filled spaces, flooding = pathology.

Key agents are:

- Water molecules: Based on possibility and dynamics of extravasation.
- Macromolecules: Plasma proteins affecting *OP*.
- Erythrocytes: For visual realism and also for future extensions (e.g., modeling oxygenation).

It is run in two scenarios:

- Normal: physiological equilibrium ($HP = 18$ mmHg, $OP = 25$ mmHg), when extravasated water is drained by lymph drainage so that alveolar flooding is prevented.
- CPE: pathology ($HP = 21\text{--}24$ mmHg, $OP = 24$ mmHg), water exceeds ACM's 400-mL capacity, inundation.

The main output is the volume of fluid outside of blood vessels (graphs). The model interface has a scenario picker and sliders, which are set to values in the scenario but can be adjusted for investigation. Figure 2.1 shows the interface, physiology environment, and agent interactions.

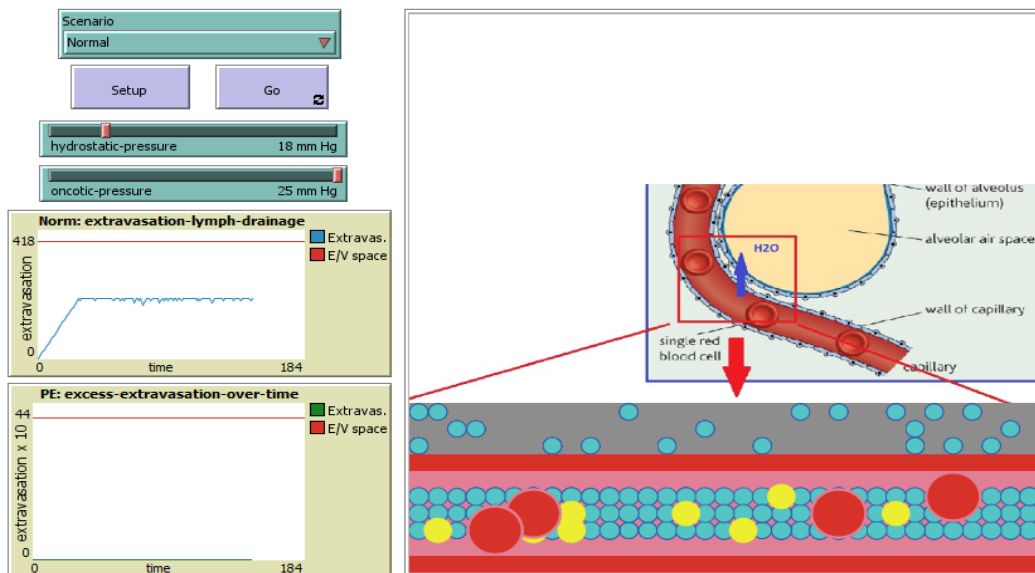


Figure 2.1. General view of the model interface in the “Normal” scenario. [69]

The scenario chooser (top left) selects the simulation mode. The sliders below are used to set the simulation mode. The upper plot shows lymphatic drainage (blue line), and the lower plot tracks extravascular water (green line) against the 400-ml threshold (red line). The right window

depicts alveoli (white), ACM (grey), and capillaries (red/pink) with agents: water molecules (blue), macromolecules (yellow), and erythrocytes (red).

The model reduces Starling equation [70]:

$$Q = L_p S ([P_c - P_i] - \delta [\pi_p - \pi_i]), \quad (2.9)$$

where Q is the flux, $L_p S$ is the filtration coefficient depending on permeability of membrane to fluid, P_c and P_i are hydrostatic pressure (in capillary space and interstitial space), δ is reflection coefficient describing ability of membrane to prevent protein movement, π_p and π_i - oncotic pressure (plasma and interstitial) controls the resulting fluid flux (Eq. 2.8).

Normally $OP > HP$, retaining fluid inside intravascular space or drained by lymphatic drainage. In CPE, hydrostatic pressure HP increases (e.g., heart failure), so that fluid flows into extravasation. Lymphatic drainage capacity is about 400 mL, which is the overflow limit (clinical observations of CPE).

2.4.2. Dynamics of the Model

Agent interactions and environment rules control dynamics of model:

- Water Movement: water molecules flow right in capillary simulating blood flow and extravasate into ACM based on $(HP - OP)$.
- Lymphatic Drainage: removes excess fluid and mimics physiologic clearance.
- Plot Outputs: top graph is lymphatic outflow (blue line); bottom graph is extravascular fluid (green line) versus 400 mL (red line).

The design is educational, open, and clinically relevant but does not consider HP/OP interaction or complexity such as permeability change or interstitial pressure changes. Erythrocytes are decorative but provide a foundation for future extensions, such as modeling gas exchange.

2.4.3. Implementation of the Cardiogenic Pulmonary Edema Model

The model was implemented in NetLogo 6.2 because it was easy to use, and built-in graphing and spatial agent-agent interactions are supported. Iterative procedures for initializing the physiological environment and running simulations [69] are available for research and instruction purposes.

Setup Procedure

The setup procedure initializes the simulation by:

1. Clearing the NetLogo world to reset all parameters.

2. Creating the physiological environment using patch colors: white for alveoli, grey for ACM, and red/pink for the capillary and its walls.
3. Importing an alveoli image to enhance visual fidelity and contextual understanding.
4. Agents to be generated: water molecules, macromolecules, and erythrocytes, inside capillary to simulate blood flow.

This configuration can be initialized consistently using simulations where agent distributions are physiologically realistic. The use of patch colors and imported images has significantly improved the overall quality. This approach is crucial for educational applications that require user-friendly interfaces.

2.4.4. Simulation Dynamics

The *go procedure* runs the simulation over and over again:

- Agent Movement: Water molecules, macromolecules, and erythrocytes move right in the capillary, simulating blood flow. The extravasation method controls the pressure differential that pushes water from the capillary into ACM.
- Lymph Drainage: The Drain-ACM function drains extra water from ACM patches if the volume is below the threshold of 400 mL (natural clearance mechanisms).
- Output Tracking: Plots are dynamically updated in real time. The blue line in the upper plot shows whether the system can drain fluid (lymphatic drainage). The lower graphic shows extravascular water (green line) versus the 400-mL threshold (red line): if flooding occurred or not.

We wanted the model to be accessible to medical students and clinicians, so simple implementation was preferred but deep enough for CPE dynamics. Erythrocytes are ornamental but allow future developments such as oxygenation or gas exchange. NetLogo is open source, making it easy to share and adapt models for research and education.

2.4.5. Technical Problems

The model was written in NetLogo 6.2 and uses its strong agent-based modeling capabilities. Built-in plotting on the platform provides real-time output, which is important when students need feedback immediately (e.g., in educational applications). HP and OP sliders let users explore the effects of parameters interactively, increasing the model's exploratory value. There are two versions of the model: the experimental version and the demo version that can be downloaded from online sources [63, 66].

2.5. Cardiogenic Pulmonary Edema Model Results

To illustrate how extravascular lung water grows with *HP* and *OP*, we run the CPE model under two main conditions: “Normal” and “Cardiogenic-Pulmonary-Edema“. We also performed some tests to assess parameter sensitivity by changing *HP* (see [69] for details). Simulations ticks (time intervals), water molecule counts, and graphical outputs are reported. The extravascular space is set to 400 mL.

2.5.1. Normal Scenario

Simulations have been started with water molecules, macromolecules, and erythrocytes in capillary under normal *HP* ($HP = 18$ mmHg; $OP = 25$ mmHg). Water extravasation to ACM was regulated during 30 ticks and reached 200 mL (for simplicity 1 molecule = 1 mL). The blue line of lymph drainage in upper graph increased steadily and levelled off eventually as clearance below 400 mL was ensured. The lower plot has moderate accumulation with level green line (Figure 2.2). This is physiologically reasonable because *HP-OP* reduces net fluid flux and keeps the lung dry when modest extravasation occurs.

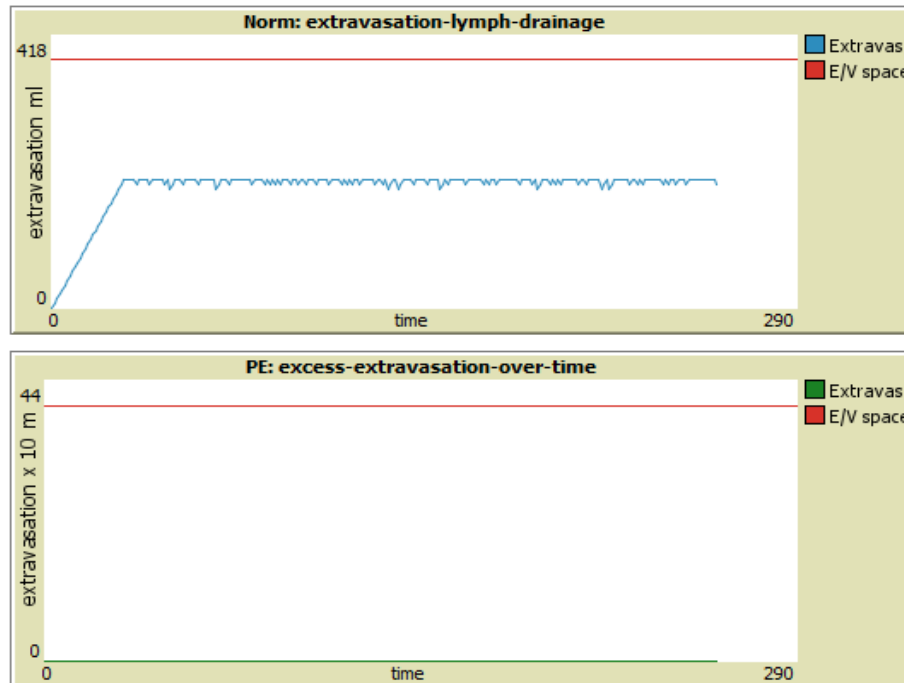


Figure 2.2. Plot view in “Normal” case. Upper graph: Lymphatic outflow (blue line). Lower graph: Extravascular water (green line). 400-mL limit (red line). “Extravas.” = volume extravasated. E/V space = ACM capacity). [69]

The Normal case shows how well the model mimics physiological equilibrium as lymph drainage eliminates minor extravasation (clinical observation of healthy lungs where fluid homeostasis prevents alveoli flooding [59]). Visual output with clear plots and agent interaction makes abstract ideas concrete.

2.5.2. CPE Scenario

In CPE scenario ($HP = 21\text{--}24$ mmHg, $OP = 24$ mmHg), fluid leakage increased rapidly (by $\sim 10\text{--}30$ ticks) over the 400-mL threshold and flooded ACM and alveoli too. In Figure 2.3, plots for $HP = 23$ mmHg, the green line crosses the red line (pathological flooding as in clinical CPE from heart failure [69]); the blue line in upper plot shows lymphatic drainage cannot keep up with rapid extravasation and collapse of system under pathological conditions.

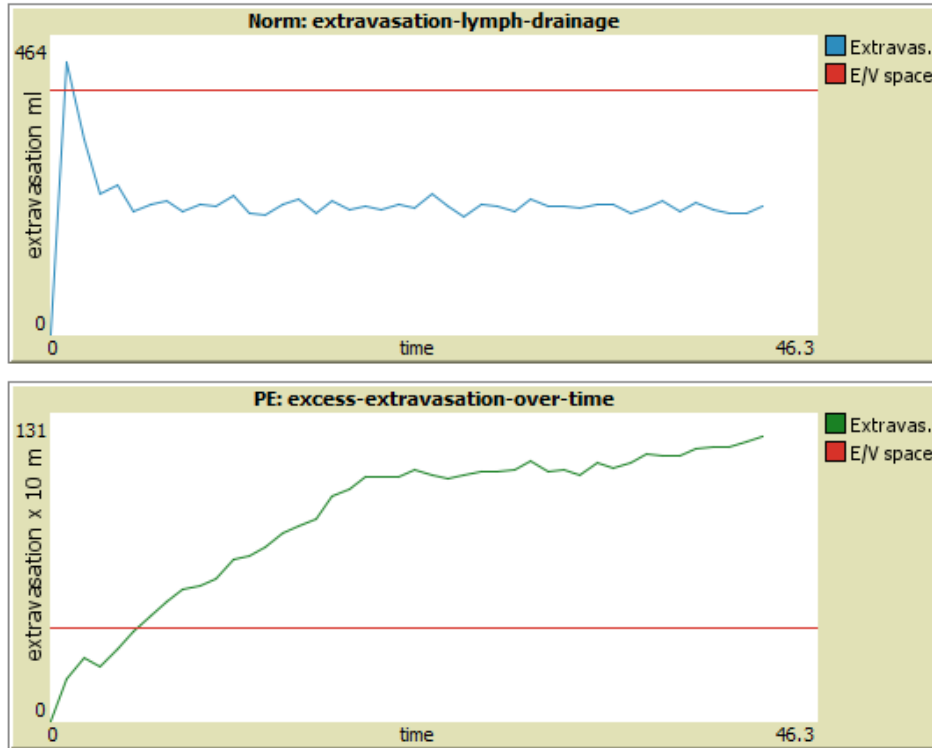


Figure 2.3. Plot view in “Cardiogenic-Pulmonary-Edema” scenario: Top plot: Lymphatic drainage (blue line); Bottom plot: Extravascular water (green line) crossing 400 mL threshold (red line). [69]

The CPE scenario represents the very rapid development of alveolar flooding characteristic of CPE, where increased extravascular water (e.g., due to left ventricular dysfunction) overwhelms clearance mechanisms. Model capability to show this changing mechanism stresses its utility as a clinical tool to predict PE onset and timing of intervention.

2.5.3. Parameter Sensitivity

To test the sensitivity of the model, we oscillated HP between 21 and 24 mmHg (5 attempts for each level of 40 ticks, 1 mmHg step in HP) with BehavioralSpace tool in NetLogo (we averaged results at each tick to look for trends). As shown in Figure 2.4, a higher HP leads to

earlier (grey dotted line) exceedance of 400 mL threshold and larger volume of extravascular water at 40 ticks.

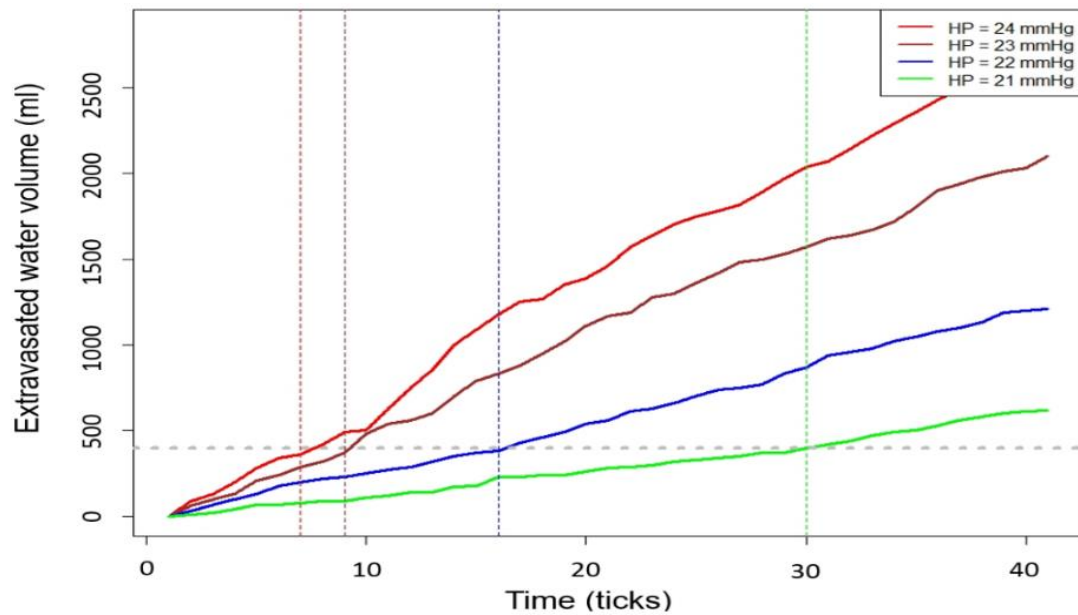


Figure 2.4. A higher *HP* leads to earlier (grey dotted line) exceedance of 400 mL threshold and larger volume of extravascular water at 40 ticks. Time course of extravascular water volume for different levels of *HP*. The grey dotted line is the 400-mL threshold (earlier crossing suggests more severe edema). [69]

Statistical analysis with one-way ANOVA was performed on two metrics: time to exceed 400 mL (*time_to_exceed*) and volume at 40 ticks (*value_at_40*), and p-values < 0.05 showed the *HP* effect on edema formation significance. Pairwise comparisons (e.g., Tukey HSD) may be more granular but were not used here. These results demonstrate the exploratory value of the model: users can quantify how a change in parameters affects pathological outcomes (a key feature for clinical and educational applications).

2.5.4. Further Remarks

The model's results provide several insights:

- **Physiological realism:** Normal stability and quick flooding of CPE scenario match clinical observations where *PCWP* > 18 mmHg indicates risk for CPE [71]. The simplified Starling equation in the model describes well interaction that governs fluid flow.
- **Educational utility:** The model is useful for instruction because situations shown by plots and agent motions are clearly separated. Students can see how small increases in *HP* (e.g., from 18 to 21 mmHg) move the system from equilibrium to pathology.

- **Therapeutic Relevance:** Sensitivity analyses showed that played a major role, so the model might predict important thresholds for intervention (e.g., diuretics to lower *HP*), which is consistent with therapeutic approaches used to treat CPE in heart failure patients.

The model is available in demonstration and experimentation versions on web repositories [63, 66]. The versions provide guided learning and open-ended exploration, respectively, which increase usefulness of model in critical care education and research.

2.6. Cardiogenic Pulmonary Edema Model: Clinical and Educational Potential, Limitations, and Future Directions

The ABM of CPE provides an example of how hydrostatic and oncotic pressures interact to promote extravascular lung water build-up (see Figure 2). It helps us understand physiological balance and pathological processes in critical care. This section describes the clinical and educational potential, limitations, and future directions of the model, highlighting its function as advancing complex empirical applications in medicine.

2.6.1. Applications in Clinical Medicine

The model can simulate CPE development with left ventricular failure and *PCWP* >18 mmHg [69]. Clinicians could input live data (*PCWP* for hydrostatic pressure, serum albumin for oncotic pressure) into the model to predict pulmonary edema in patients at risk (heart failure or fluid overload), e.g., *PCWP* being the hydrostatic pressure and CPE scenario being above 400 mL in 10-30 ticks (*HP* < 19 mmHg, needing diuretics/vasodilators). Unlike equations, ABM has a dynamic feature so “what if” analysis can be done in real time, improving ICU triage decision-making.

New technologies enhance the clinical effectiveness of the ABM concept. Patient-specific computation models (especially cardiology using computational fluid mechanics [72]) give us a way to integrate this ABM with wearable sensors or electronic health data. The algorithm could track continuous albumin levels or pulmonary wedge pressure, so that real time PE risk predictions are delivered. While not clinically validated, the model’s response to *HP/OP* change is a proof of concept for specific respiratory treatment. Chest imaging or clinical outcome assessment in the future could show its prognostic ability and allow its use in ICU settings to improve patient management.

This model can be used by medical students for interactive model simulations for scenario planning because it simulates shifts in parameters (e.g., a clinician may want to present results on how different doses of a diuretic affect extravascular fluid accumulation). It is an agile approach compared to static CPE models that do not capture temporal and spatial details. ABM provides a

scalable platform for developing decision support tools tailored to intensive care challenges (biomedical engineering vs. clinical practice).

2.6.2. Educational Value

Interactive model simulation can be used by medical students to study CPE pathophysiology. The difference between physiological equilibrium (adequate lymph drainage) and pathological states (alveolar inundation) in Normal and CPE cases [70] shows *HP/OP* interaction in the Starling principle. Students can demonstrate changes in extravascular water instantaneously by changing *HP* (e.g., from normal to PE-like inundation) using interactive sliders. Simulation-based learning increases knowledge of complex ideas better than traditional methods [64]. The blue drainage graph and green accumulation graph are abstract criteria (e.g., 400 mL).

The model is simple enough for novices to understand (it only has *HP/OP* and ignores other variables), but because it can be extended (e.g., oxygenation added), it is better suited for more advanced learners. ABM dynamic agents (e.g., water molecules flooding alveoli) make learning vivid and effective rather than static schematics; NetLogo being open source also allows instructors to modify the model for wider respiratory courses (simulations of non-cardiogenic pulmonary edema or therapeutic effects, etc.). Application links theory to practice: doctors learn fluid mechanics in critical care from theoretical physiology.

The visual and interactive nature of the model also enables interdisciplinary education (biomedical engineers might study computational aspects of ABM while medical students concentrate on clinical implications). Students think about patient care holistically by using CS principles and supporting their understanding of complex systems. Its flexibility allows educators to tailor simulations to specific learning goals (e.g., hypoalbuminemia effects on OP or ventilator settings on lung dynamics).

2.6.3. Limitations and Future Research

The model has strengths but also limitations that need to be considered: *k* permeability is a constant membrane with no interstitial pressures, so real complications are excluded; oxygenation (main CPE output) is omitted; and non-cardiogenic components (ARDS) are excluded; oxygenation (main CPE output) is not accounted for; and non-cardiogenic components (ARDS) are not included. Relate to clinical data (e.g., extravascular lung water vs. thoracic imaging or pulmonary artery catheter measurements). Animal models or retrospective clinical data could provide additional validation to check the model's relevance to real life.

These limitations might be overcome by future improvements:

- When working with O₂/CO₂ gasses: To study impairment of gas exchange which is a major feature of CPE.
- Variable Permeability: ACM permeability can be modelled as varying to simulate non-cardiogenic PE or therapeutic effects (e.g. corticosteroids).
- Treatment Simulations with diuretics or ventilatory assistance to study effect of treatment on fluid flow.
- Real-time information integration: Integrate the model with patient-specific data (wearable sensors or electronic health records) for customized forecasts.

These extensions would increase the clinical and educational impact of the model and make it an adaptable tool for intensive care research and training. Clinicians and data scientists could support validation and integration to ensure the model meets ICU use demands (outside the scope of the present parameters, but these data show the model capability and spur further development in biomedical engineering and medical training).

2.7. Synthesis of Chapter 2

In this chapter we discuss CS and ABM in intensive care medicine and their potential for managing critical illnesses because they are complex and unstable by nature. CS is an overarching view of complex adaptive systems with nonlinearity, emergence, adaptation, and self-organization as common features (sepsis, ARDS, CPE, etc.). ABM is a computational system from CS that models interactions between individual agents to generate emergent responses and provides a new way to model physiological, clinical, and operational obstacles in the ICU.

In the background, CS development was part of interdisciplinary research at the Santa Fe Institute, and ICU is where nonlinear connections are most important to outcomes. ABM's ability to encode heterogeneous agents and spatial dynamics makes it especially well-suited for describing complex medical systems, such as biological processes like cytokine cascades in sepsis and ICU workflows like ventilator allocation; however, despite computational challenges and validation issues, ABM has transformational potential for customized healthcare and resource planning when used with real-time data.

We give the mathematical description of ABM. Updates of agent states, interaction chances, and metrics of emergent behavior are given. The simplified Starling equation, $Q = k(HP - OP)$, shows how physiological principles are translated into rules of computation that allow active simulations of fluid shifts in CPE. The formalism allows stochasticity and spatial differences, which are important for representing patient-specific responses in intensive care.

As a case study, we implement CPE's ABM in NetLogo (Figure 2.1). The model abstracts CPE as a dynamic process caused by hydrostatic and oncotic pressures: fluid accumulation in a physiological space (capillaries, ACM, and alveoli). Normal hydrodynamics show balanced flow where drainage keeps the avascular water below 400 mL (Figure 2.2); CPE shows rapid flooding when it exceeds this value as clinical heart failure does (Figure 2.3). Parameter sensitivity experiments with tools like NetLogo's BehavioralSpace (Figure 2.4) show that edema formation is important and statistically significant (ANOVA, $p < 0.05$).

The model is a clinical prototype for decision support tools that predict PE progression by simulating real-time changes in parameters. It is sensitive to change, so "what-if" scenarios can be used for triage in ICUs and for planning interventions. Predictive performance could be enhanced further if wearable sensors or electronic health records are combined to allow individualized respiratory care. From an educational perspective, the interactive interface with sliders and real-time plots helps medical students understand fluid mechanics, linking theory to intuition in clinical practice. The accessibility of the NetLogo platform allows implementation in medical curricula, but non-cardiogenic PE or treatment simulations may be added in the future.

The limitations (simplified permeability; no oxygenation) are things to work on. Oxygenation agents, variable permeability, or treatment simulations could be added (validated against clinical data); these would increase the model's clinical and educational value so it becomes a useful tool for intensive care research and training.

This ABM demonstrates how complexity science (CS) can enhance intensive care by modelling the intricacies of physiological and clinical systems.

3. SYSTEM DYNAMICS MODELING IN COMPLEXITY SCIENCE: APPLICATIONS TO INTENSIVE CARE MEDICINE (HUMAN CIRCULATORY SYSTEM)

3.1. Complexity Science and System Theory in Medicine (ICU). A System Dynamics Approach

Complexity science (CS) and systems, when combined with real-time patient data, represent a major step towards accuracy and clinical relevance (intensive care: rapid change requires continuous monitoring), since static models become dynamic tools that adapt theory (ST) and are paradigm shifts away from the reductionism that has dominated biomedical research and clinical practice for decades. Reductionism is useful because it allows us to isolate specific mechanisms (e.g., molecular pathways or organ functions), but it cannot explain the whole of human health and disease (especially in ICUs, where multiple systems interact unpredictably). CS (from physics, biology, and computer science) views biological and healthcare systems as CASs - systems where many components interact dynamically and nonlinearly so emergent behaviors can't be predicted from parts alone [1].

Adaptation - systems change with changing conditions; self-organization—order spontaneously forms without central control; emergence - novel properties arise from interactions; no rigid hierarchy allows fluid distributed decision-making [11].

These ideas are relevant in the ICU because patients often present with multisystem organ failure (cardiovascular, respiratory, renal, neurological, and immunological processes) [48].

In the ICU setting, CS provides us a lens through which to view the unpredictable trajectories of critically ill patients. For example, sepsis is a life-threatening disease common in ICUs, where the host response to infection is dysregulated, leading to cascade inflammatory pathways that cause organ dysfunction not by linear cause-and-effect but by emergent interactions between cytokines, immune cells, and endothelial barriers. Linear models may predict outcomes based on isolated biomarkers, but CS shows how small perturbations (e.g., minor infection or fluid imbalance) amplify through feedback loops into systemic chaos (shock/multi-organ failure) [73]. Moreover, the ICU itself can be seen as a CAS - not just the physiology of the patient but also interactions between healthcare providers, technology (ventilation, monitors), protocols, and even hospital resources or family involvement [24, 73]. This situation leads to systems thinking being used to deal with complexity because decision-making is nonlinear; outcomes are built from these elements. CS provides structured frameworks for modelling and analyzing this interaction (system theory, ST).

System theory developed out of Ludwig von Bertalanffy's general systems theory in the mid-20th century and emphasizes feedback loops (positive amplification, negative stabilization), boundaries of systems, and exchanges with the environment.

In medicine, ST has been applied to conceptualize delivery of health care as interconnected systems (intensive care: closed-loop feedback from monitoring devices to therapeutic interventions). For example, in ARDS, ST helps model how ventilator settings interact with lung mechanics, oxygenation levels, and inflammatory response so leverage points for intervention are identified. More recently, CS has been combined with ST to measure complexity in ICUs (entropy, fractal dimensions, or sensitivity analysis of vital signs/laboratory values) so as to see how changes in vital signs or laboratory values follow treatments and help early detection of deterioration. This work shows that these frameworks are largely untapped for real-time predictive modeling (high dimensionality and nonlinearity of data) [24, 73].

There is also a big difference between ABM and SDM modeling paradigms within CS. ABM is often thought of as closer to the fundamental CS principles because it simulates systems bottom up—individual agents (cells, molecules, patients, or clinicians) following local rules generate emergent behavior at the macro level [74]. In ABM agents are heterogeneous, spatially organized, and adaptive and learnable, which makes them well suited to stochastic processes and emergence in complex scenarios (e.g., infectious disease outbreaks in hospitals; immune responses in critical illness). For ICU applications, ABM can model nurse staffing dynamics (individual agents (nurses) interact with patients and equipment) to see how local decisions aggregate into system efficiency or bottlenecks [73].

ABM is good at modeling discrete events and spatial heterogeneity, so “what if” scenarios involving individual variability can be explored in detail [75].

SDM works top-down, aggregating the data using stocks (accumulations), flows (rates of change), and feedback loops to represent behavior without explicitly modelling agents. ABM is good at discrete heterogeneous entities, but SDM excels at continuous processes, time delays, and nonlinear accumulations common in physiological and organizational systems [76]. Other advantages of SDM in complex systems are ease of calibration from aggregate data, smaller-scale simulations required for large-scale simulations, and better visualization of causal relationships through loop diagrams, which clinicians and policymakers use to communicate. In the ICU, SDM is especially useful for applications that are often underutilized, such as resource allocation during surges (e.g., pandemics) or managing chronic disease progression, where individual data may be unavailable or too noisy, making aggregate trends sufficient for decision-making [73].

One of the building blocks of SDM in physiology is "lumped parameters model" (LPM): distributed systems are broken down into discrete compartments with averaged properties (simplification without loss of essential dynamics) [77, 78]. In cardiovascular modelling blood vessels are treated as electrical analogs (resistors for friction, compliances for elasticity, and inertances for inertia), and spatial details are aggregated into zero-dimensional elements so that hemodynamics can be simulated quickly without having to use complicated partial differential equations needed by distributed models [66]. LPMs are invaluable in ICU simulations (heart failure with preserved ejection fraction HFpEF; compartments represent heart chambers and vascular segments to predict pressure-volume loops and cardiac output under stress). By using patient-specific parameters from non-invasive measurements, LPMs combine theory with clinical utility (shock, congenital heart defects, etc.), which abstracts distributed systems into discrete compartments with averaged properties, simplifying analysis without sacrificing essential dynamics [77, 78]. In cardiovascular modeling, LPMs treat blood vessels as electrical analogs—resistors for friction, compliances for elasticity, and inertances for inertia—aggregating spatial details into zero-dimensional elements. This method makes it possible to simulate hemodynamics quickly without having to use the complicated partial differential equations that distributed models need [66]. LPMs are invaluable in ICU scenarios, such as simulating heart failure with preserved ejection fraction (HFpEF), where compartments represent heart chambers and vascular segments to predict pressure-volume loops and cardiac output under stress. By incorporating patient-specific parameters from non-invasive measurements, LPMs bridge theoretical modeling with clinical utility, aiding in conditions like shock or congenital heart defects [78].

Another CS/ST combination is the “digital twin” concept: virtual replicas of physical systems for real-time simulation and prediction in personalized medicine. In healthcare, digital twins use data from wearables, EHRs, and imaging to replicate a patient's condition, allowing for the testing of treatment scenarios and outcomes; for example, in intensive care, they can optimize ventilator settings for ARDS or medication dosing for sepsis by employing CS elements such as adaptation and emergence to predict patient trajectories. However, challenges include privacy issues, the need for model validation against real-world variability, and the integration of AI to enhance accuracy. Digital twins simulate myocardial infarction effects for cardiovascular applications, which create personalized treatments and lower risks in high-stakes ICU settings.

Beyond these, CS also works with machine learning to develop hybrid models that learn patterns from data while preserving dynamics (population-level ICU modeling). In under-explored areas like SDM for intensive care, these approaches will revolutionize care by tackling complexity head-on and ultimately improve outcomes through proactive systems-informed strategies [79-86].

3.2. General Mathematical Formalism Behind System Dynamics Modeling

System dynamics modeling (SDM) is based in a rigorous mathematical formalism that enables modeling and investigation of the temporal dynamic, including ones with nonlinear behavior [87]. The ODEs at its core are ordinary differential equations that describe how stocks (the accumulation of quantities such as population, resources, or, in medicine, blood volume or drug concentrations) change over time due to inflows and outflows. The core equation for the stock $S(t)$ is:

$$\frac{dS(t)}{dt} = Inflow(t) - Outflow(t). \quad (3.1)$$

The rate of change equation can be integrated to express the stock as an accumulation:

$$S(t) = S(0) + \int_0^t [Inflow(\tau) - Outflow(\tau)] d\tau. \quad (3.2)$$

Inflows and outflows are functions that depend on the level of the stock, other variables, constants, or time, so dynamic interactions can be represented.

Feedback loops are central to SDM; positive (reinforcing growth/decline) and negative (balancing stability). A simple positive feedback model (population growth) is:

$$\frac{dP}{dt} = rP, \quad (3.3)$$

with the solution $P(t) = P_0 e^{rt}$ illustrating exponential amplification. Negative feedback (e.g., thermostat regulating temperature) could be written as:

$$\frac{dT}{dt} = -k(T - T_{set}), \quad (3.4)$$

yielding exponential decay toward equilibrium. State-space formalism for multivariable systems gives a matrix representation of the system dynamics:

$$\dot{x} = Ax + Bu \quad (3.5)$$

$$\dot{y} = Cx + Du, \quad (3.6)$$

where x is the state vector (e.g., stocks), u - inputs (external influences), y - outputs, and matrices A, B, C, D define system dynamics. Nonlinearities are captured by quadratic or product terms (e.g., in predator-prey models like Lotka Volterra):

$$\frac{dx}{dt} = \alpha x - \beta xy, \quad (3.7)$$

$$\frac{dy}{dt} = \delta xy - \gamma y. \quad (3.8)$$

Oscillatory behavior in ecological or physiological systems can be captured here: two populations are denoted x and y , and α , β , δ , and γ are parameters of these populations [88]. This approach is used to model oxygen consumption and delivery in ICUs.

Delays are common in biological processes (drug absorption, etc.), so delay functions are used to model them. First order delay approximates distributed lags:

$$Delay(t) = \frac{1}{\delta} \int_{t-\delta}^t f(\tau) d\tau, \quad (3.9)$$

where δ is the delay time. Numerical solution methods, such as Euler's method, are needed to solve ODEs, as they provide approximate solutions:

$$x_{n+1} = x_n + hf(x_n, t_n) \quad (3.10)$$

and higher order Runge-Kutta methods for stiff systems.

Auxiliary variables and constants allow the model to be flexible, such as density-dependent rates:

$$Rate = k \cdot \frac{Stock}{Capacity} \quad (3.11)$$

for logistic growth limits; sensitivity analysis looks at how parameters affect the response (e.g., partial derivatives or Monte Carlo simulations). In medicine this formalism allows scenario testing (e.g., epidemic spread modeling vs. physiological responses) so that policy design and intervention can be optimized.

3.2.1. Basic Components and Mathematical Aspects of SDM in the Circulatory System

The human circulatory system is a classic example of a complex adaptive system that can be modeled using SDM [89] because it has the heart, blood vessels, and blood all interconnected by feedback loops that regulate physiological homeostasis. Static models (anatomical diagrams or equilibrium equations) typically cannot capture time-dependent interactions and nonlinear responses that are typical for cardiovascular dynamics under pathological conditions (e.g., hypertension or heart failure). SDM overcomes these limitations by using stocks to represent accumulations (e.g., blood volume in different compartments), flows to represent rates of change (e.g., blood flow between arteries and veins), and feedback loops to model regulatory mechanisms (positive loops may amplify changes, as in increased heart rate during exercise to increase cardiac output; negative loops stabilize the system as baroreceptors adjust blood pressure).

The basic elements of SDM here are:

A. Basic Components

1. Stocks Model Dynamics through Stocks and Flows

- Stocks are accumulations within the system (e.g., volume of blood in the left ventricle or aorta).
- Flows are rates at which stocks change (inflow from pulmonary veins or outflow to systemic circulation).

2. Feedback Mechanisms (Loops)

- Positive Feedback: Promotes deviations (sympathetic stimulation increases heart rate in response to low pressure);
- Negative Feedback: Suppresses changes (renin-angiotensin system controls blood volume and pressure).

B. Primary Mathematical Formulations

a. Formulations concerning Stocks and Flows

- Alteration in Fluid (Blood) Volume (V):

$$\frac{dV}{dt} = R_{in} - R_{out}, \quad (3.12)$$

where R_{in} denotes influx of blood into compartment and R_{out} efflux from it.

- Cardiac Output (CO):

$$CO = HR \times SV, \quad (3.13)$$

where HR is heart rate measured in beats per minute, and SV is stroke volume measured in mL.

b. Feedback Mathematical Formulations

- Blood Pressure (BP) Rule:

$$BP = CO \times TPR, \quad (3.14)$$

where TPR is total peripheral vascular resistance measured in mmHg·s/mL.

- Baroreceptor Reflex: Modifies heart rate and vessel tone:

$$HR = f(BP), \quad (3.15)$$

where f is the response function (nonlinear usually since physiological thresholds are reached).

C. Modeling and Analysis

Differential equations model time variations, which can be solved numerically to study responses to disturbances; in a lumped parameters model (LPM), the circulatory system is divided into compartments such as the aorta (compliance) and systemic resistance so hemodynamics can

be efficiently computed [90]. Formalism captures interplay between components, so myocardial infarction (MI) simulations (decreased myocardial contractility disrupts feedback loops leading to reduced CO and compensatory increases in HR) can be done.

The possibilities for SDM of the circulatory system are vast due to the numerous interactions involved, such as physiological states (e.g., normal resting vs. pathological conditions like atherosclerosis that increase resistance), time-varying interactions (e.g., changes in heart rate and blood volume), hypotheses regarding interventions (e.g., lifestyle changes that reduce risk or medications that alter contractility), and the ability to personalize models for individual patients to predict disease progression or treatment efficacy.

Benefits are that SDMs can show hidden interactions, leverage points for intervention success, and educational tools (medical training: simulations illustrate cardiovascular physiology). In practice, SDM guides public health policy by modelling the population effects of preventive measures, implementing clinical decision systems with dynamic risk assessments, and facilitating pharmaceutical development, such as simulating drug effects on hemodynamics to reduce trial costs.

LPMs go beyond this in that they aggregate distributed properties (e.g., treat the whole arterial tree as a single compliant vessel), which is computationally tractable for ICU use; MI LPMs predict how infarct elastance changes affect PV loops (e.g., by inotropes); and CS principles make models more robust because emergence is accounted for in system responses.

3.3. Real-Time Data Integration into SDM and Model Generation

Integrating real-time data into system dynamics models (SDM) data represent a major step towards accuracy and clinical relevance (intensive care: rapid change requires continuous monitoring), since static models become dynamic tools adapting to individual variability from wearable devices, sensors, and EHRs to update parameters in real time. Key components include data acquisition, processing and model adaptation, technical infrastructure, feedback and decision support, and challenges.

A. Data Assimilation Strategies

1. Data Retrieval: Wearables (smartwatches, continuous glucose monitors) and ICU sensors measure HR, BP, oxygen saturation, and ECG waveforms wirelessly.

2. Information Handling: Algorithms tune and condition data to remove noise, artifacts, or deficiencies (Kalman filtering, moving averages, etc.). ML algorithms are often used for this.

3. Model Adaptation: ML algorithms (neural networks/Bayesian updating) adapt model parameters (resistance, elastance) dynamically with incoming data so that they match the current state of the patient.

B. Technical Infrastructure

1. Provision of computational resources online: AWS or Azure provide expandable storage and compute power for large datasets and complex simulations without local hardware constraints.
2. APIs and interoperability: Standardized APIs allow devices, EHRs, and SDM software to exchange data (FHIR is one example of a protocol ensuring compatibility across systems).

C. Feedback and Decision Support

1. Data Visualization Dashboard: User-friendly interfaces show model results alongside patient data so that healthcare professionals can see changes in cardiac output, blood pressure, or volume.
2. Alerts and Notifications: Automated systems warn of predicted adverse events (e.g., impending hypotension from MI simulation).
3. Customized Treatment Modifications: Insights inform real-time changes to treatment (e.g., change vasopressor dosage or hydration management according to model predictions).

D. Obstacles and Factors to Consider

1. Privacy and Security Aspects: Standards like HIPAA require encryption, access control, and anonymization to protect sensitive data [91].
2. Model validation: Ongoing comparison with clinical outcomes improves accuracy (cross-validation or sensitivity analysis, etc.).
3. User education: Healthcare professionals need training to understand results and incorporate them into their processes (overdependence).

This research [90, 92, 93] used NetLogo to build SDM models of the human circulatory system. NetLogo is a user-friendly platform for stocks, flows, and parameter simulations. Figure 3.1 shows the system dynamics scheme of the last version of the model (six compartments (stocks): heart chambers and vascular segments; over 30 parameters (diamonds); influences (thin arrows); and flows (thick arrows)). The mathematical equations governing these interactions have been given previously, and for this model the main parameters can be generally represented as follows:

1. Volume (ventricular, arterial, or venous segments):

$$\frac{dV_{c_i}}{dt} = \frac{(P_{c_{i-1}} - P_{c_i})}{R_1} - \frac{(P_{c_i} - P_{c_{i+1}})}{R_2}, \quad (3.16)$$

where V is volume, P is pressure and R is resistance; every three formally consecutive compartments (e.g., pulmonary veins, left ventricle and aorta) are denoted by c_{i-1}, c_i, c_{i+1} and resistances between them are R_1 and R_2 .

2. Pressure (vascular segments):

$$P_{c_i} = E_{c_i} \times V_{c_i}, \quad (3.17)$$

where E is elastance, V is volume, and c_i denotes the respective compartment (i.e., vascular segment).

3. Ventricular pressure:

$$P_v = tve \times E_{es_v} \times (V_v - V_{d_v}) + (1 - tve) \times P_{0_v} \times (\exp \exp (\lambda_v (V_v - V_{0_v})) - 1), \quad (3.18)$$

where $tve = e^{-80 \times (\text{time} - 0.27)^2}$ is time-varying elastance, E_{es_v} is end-systolic elastance, V_v is left ventricle volume, V_{d_v} is unstressed volume, P_{0_v} is zero-volume pressure, λ_v is the curvature of the end-diastolic pressure-volume relationship, and V_{0_v} denotes zero-pressure volume.

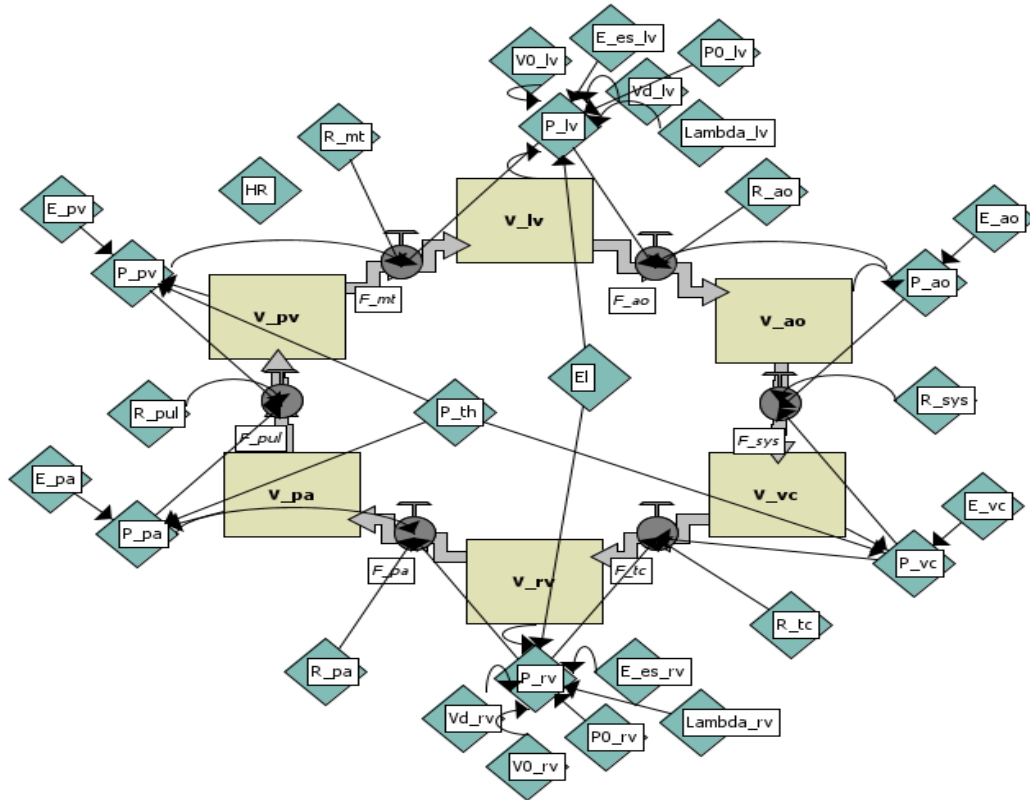


Figure 3.1. The system dynamic scheme of the created model. Squares represent stocks, and diamonds denote other various parameters. [90]

In Figure 3.1, capital letters point to a certain parameter: V – volume, E – elastance of different compartments of the system, etc.

Initial conditions (for the “Normal” scenario) are presented in Table 3.1.

Table 3.1. Initial conditions for the model

Name of the parameter	Notation	Value	Units
Systemic resistance	R_{sys}	1.0889	$\text{mmHg} \cdot \text{s} \cdot \text{mL}^{-1}$
Aortic elastance	E_{ao}	0.6913	$\text{mmHg} \cdot \text{mL}^{-1}$
Pulmonary artery resistance	R_{pulm}	0.018	$\text{mmHg} \cdot \text{s} \cdot \text{mL}^{-1}$
Intrathoracic pressure	P_{th}	-5	mmHg
Pulmonary artery elastance	E_{pa}	0.369	$\text{mmHg} \cdot \text{mL}^{-1}$
Mitral valve resistance	R_{mt}	0.0158	$\text{mmHg} \cdot \text{s} \cdot \text{mL}^{-1}$
Aortic resistance	R_{ao}	0.018	$\text{mmHg} \cdot \text{s} \cdot \text{mL}^{-1}$
Tricuspid valve resistance	R_{tc}	0.0237	$\text{mmHg} \cdot \text{s} \cdot \text{mL}^{-1}$
Pulmonary artery resistance	R_{pa}	0.0055	$\text{mmHg} \cdot \text{s} \cdot \text{mL}^{-1}$
End-systolic elastance of left ventricle	E_{eslv}	2.8798	$\text{mmHg} \cdot \text{mL}^{-1}$
Unstressed left ventricle volume	V_{dlv}	0	mL
Left ventricle zero-volume pressure	$P0_{lv}$	0.1203	mmHg
Curvature of end-diastolic pressure-volume relationship for left ventricle	λ_{lv}	0.033	mL^{-1}
Left ventricle zero-pressure volume	$V0_{lv}$	0	mL
End-systolic elastance of right ventricle	E_{esrv}	0.5850	$\text{mmHg} \cdot \text{mL}^{-1}$
Unstressed right ventricle volume	V_{drv}	0	mL
Right ventricle zero-volume pressure	$P0_{rv}$	0.2157	mmHg
Curvature of end-diastolic pressure-volume relationship for right ventricle	λ_{rv}	0.023	mL^{-1}
Right ventricle zero-pressure volume	$V0_{rv}$	0	mL

Table 3.1. Continued

Pulmonary veins elastance	E_{pv}	0.0073	$\text{mmHg} \cdot \text{mL}^{-1}$
Vena cava elastance	E_{vc}	0.0059	$\text{mmHg} \cdot \text{mL}^{-1}$
Aorta volume	V_{ao}	150	mL
Left ventricle volume	V_{lv}	120	mL
Pulmonary artery volume	V_{pa}	150	mL
Pulmonary veins volume	V_{pv}	300	mL
Right ventricle volume	V_{rv}	150	mL
Vena cava volume	V_{vc}	4150	mL
Heart rate	HR	70	beats per minute

Figure 3.2 shows the model's user interface, with sliders for parameter adjustment, monitors for real-time outputs (e.g., volumes, pressures, stroke volumes, and cardiac output), and plots for temporal dynamics. Capital letters denote different physiological parameters: V – volume, P – pressure, SV – stroke volume, CO – cardiac output, R – resistance, E – elastance, Lambda – a function influencing myocardial contractility, and HR – heart rate. Notations are similar to the ones in Figure 3.1).

This configuration supports MI simulation by altering parameters like elastance, linking to digital twins for personalized ICU monitoring, where real-time data refines predictions for better patient outcomes.

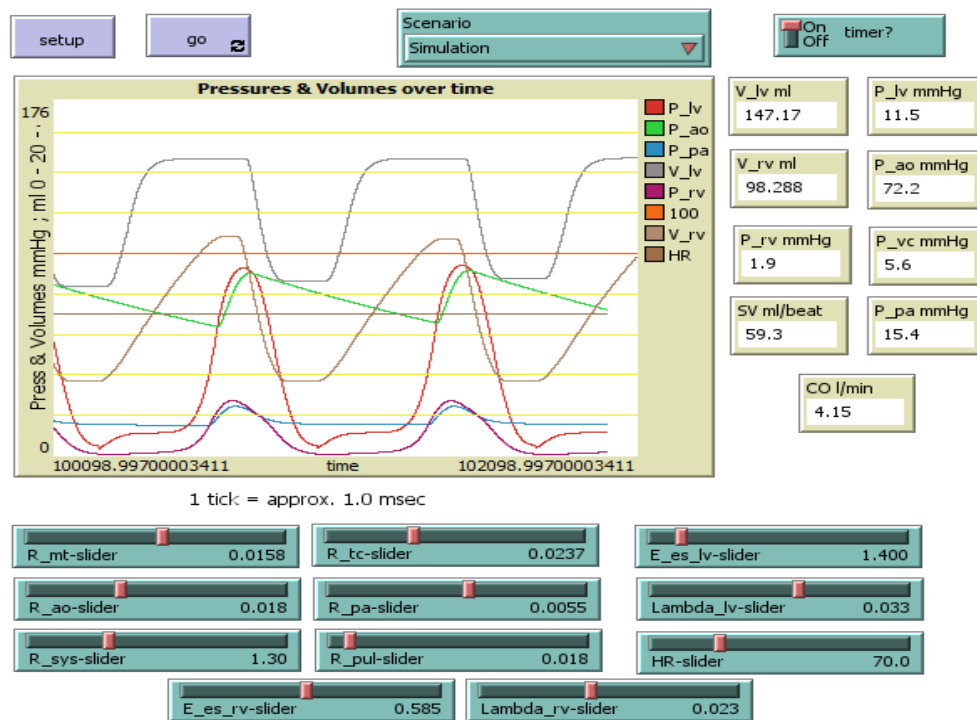


Figure 3.2. User interface of the model. [90]

3.4. Simulation results and analysis in MI

The work behind this model is still preclinical, so the core modeling aspects are well developed, but future integration with real-time data (e.g., adaptive therapies) is not yet foreseen. Simulations provide numerical data for analysis (Table 3.2): time series values of key parameters during a normal-to-MI transition reflect clinical metrics that might be used bedside.

Table 3.2. A minor excerpt of the simulation findings

LV pressure (mm Hg)	LV output (L/min)	RV output (L/min)	RV volume (mL)	AP (mm Hg)	VC pressure (mm Hg)
41.054	6.135	6.324	124.141	112.466	7.151
41.880	6.074	6.261	124.141	112.399	7.151
42.725	6.117	6.306	124.141	112.332	7.152
43.590	6.353	6.549	124.141	112.265	7.153
44.476	6.266	6.459	124.141	112.199	7.153
45.382	6.170	6.360	124.141	112.132	7.154
46.308	6.327	6.522	124.141	112.065	7.154
47.256	6.301	6.495	124.141	111.999	7.155
48.225	6.135	6.324	124.141	111.932	7.155
49.216	6.344	6.540	124.141	111.866	7.156

Note: LV – left ventricle, RV – right ventricle, AP – aortic pressure, VC – vena cava.

These values match clinical parameters so they can be incorporated into DSS. Figure 3.3 shows the pressure-volume loops (PVL) for a modeled myocardial infarction where the left ventricle (green) and right ventricle (yellow) have reduced dimensions due to filling problems.

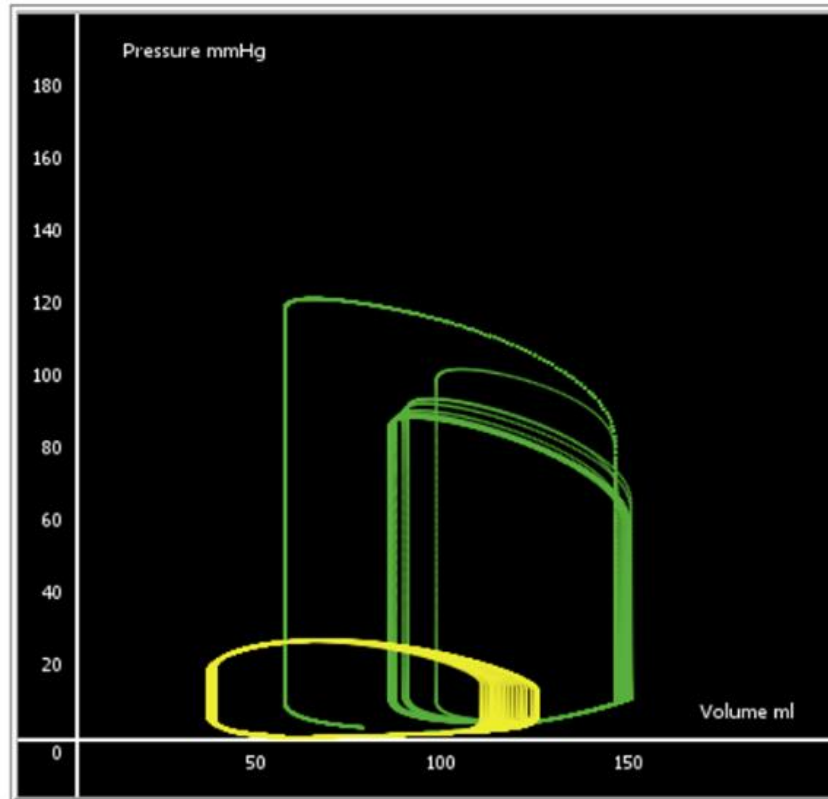


Figure 3.3. PV loops in myocardial infarction. The green color is for the left ventricle, and the yellow color – for the right ventricle. [90]

Multivariate time series analysis techniques are used for generated time series (e.g., Kolmogorov-Chaitin complexity (KCC) measures the order of the system). In figure 3.4, KCC over time is plotted for 12 parameters with a black arrow indicating MI onset; post-event complexity drops drastically, showing loss of dynamic range.

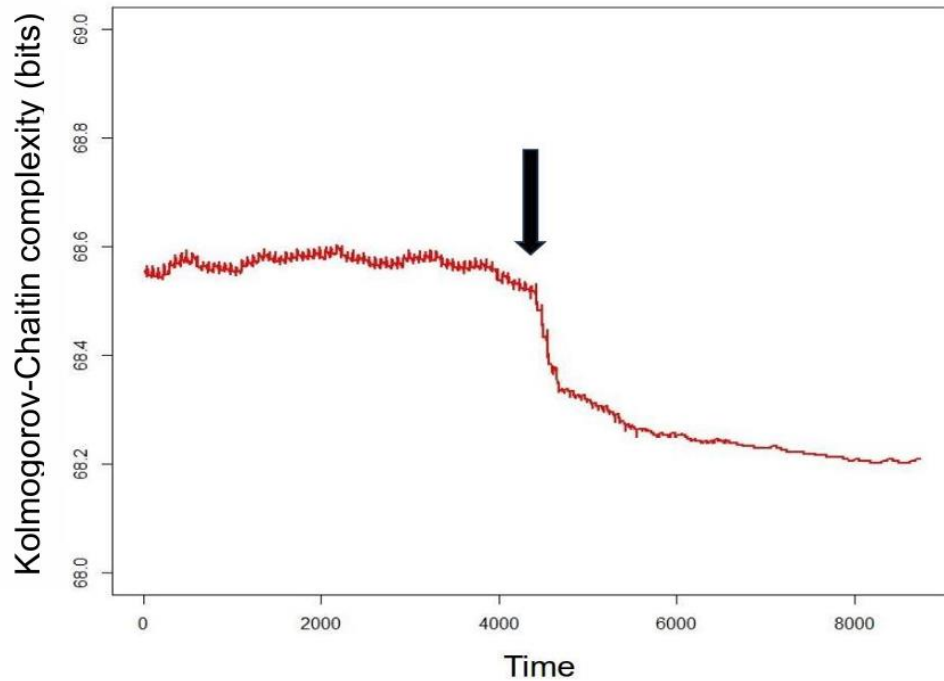


Figure 3.4. Temporal dynamics of Kolmogorov-Chaitin complexity in a simulated scenario of myocardial infarction. A black arrow indicates an event of impact (i.e., MI). [90]

Such metrics (and traditional ones like spectral analysis or chaos indicators) are complementary to traditional views (e.g., a KCC drop could be a sign of critical transition in an ICU). Building on the simulation, LPMs allow us to go into MI pathophysiology detail: how low end-systolic elastance causes forward failure, compensatory tachycardia, and backward congestion? That informs beta-blockers or mechanical support strategy. Digital twin integration personalizes this too (patient data used to predict post-MI outcomes and guide care).

3.5. Discussion and Future Directions for SDM Circulatory System Modeling

In our earlier three-compartment model, parameters such as volume status and contractility were controlled (Figure 3.1). Here we extended sliders (Figure 3.2) for all chambers, valves, and circulations to simulate finer dynamics. Experiments using NetLogo BehaviorSpace produce large amounts of data (e.g., Table 3.2 excerpts) that scale up for long-duration studies and can be selected for clinical equivalents so that real-time integration is possible.

For future real-time applications, some considerations are:

1. Real-time data acquisition: Wearables for HR, BP, and ECG (robust connectivity).
2. Data Processing: Algorithms for cleaning and dynamic parameter updates via ML.
3. Clinical Decision: Support tools (risk alerts; tailored interventions).
4. Privacy: Encryption / regulatory compliance.

SDM (especially MI) explains disease mechanisms so ICUs can predict responses and identify interventions [94]. Complexity and validation remain challenges, but synergies with AI will increase precision. Multisystem extensions (e.g., coupling to respiratory models) and clinical trials are needed to realize full potential in critical care.

3.6. Synthesis of Chapter 3

Complexity science and system theory are a departure from reductionist thinking in medicine (especially ICU patients have interconnected physiological failures). Biological and healthcare systems are complex adaptive systems where dynamic nonlinear interactions produce emergent behavior—adaptation to change, self-organization without central control, and emergence of novel properties from component interactions. In intensive care, this framework explains why patient trajectories are unpredictable (e.g., in cases of sepsis, where small changes can cascade through feedback loops into crises)—the ICU itself is a complex adaptive system that includes patient physiology, healthcare providers, technology, protocols, and the environment.

System theory builds on that concept by structuring models with feedback loops, boundaries, and exchanges with the environment (positive feedback amplifies change; negative feedback stabilizes). In intensive care these concepts explain ventilator adjustments in acute respiratory distress syndrome (ARDS) and intervention points. Entropy and fractal dimensions measure complexity; early detection of deterioration in patients with high-dimensional nonlinear data (entropy) and fractal dimension (fractal dimensions) helps.

Complexity modeling within complexity includes agent-based modeling, which simulates bottom-up emergent behaviors through individual agents (good for heterogeneous scenarios like hospital staffing or immune responses), and system dynamics modeling, which uses top-down stocks for accumulation, flows for rates, and feedback loops for dynamics without individual agents (good for continuous processes, time delays, and nonlinear accumulations; calibration from aggregate data, lower computational demands, and clear visualization of causal relationships). In intensive care system dynamics modeling helps us figure out how to use resources when there is a surge or when chronic disease is getting worse.

A tool in physiological system dynamics is the lumped parameter model, which breaks down distributed systems into compartments with averaged properties. Simulating hemodynamics becomes easier (in intensive care shock or heart failure) because of these models; digital twins create virtual patient replicas for real-time predictions that incorporate adaptation and emergence to optimize treatments; hybrid models (system dynamics + machine learning) improve predictive accuracy at the population level for intensive care modelling.

Mathematical basis of system dynamics modeling: ordinary differential equations describing changes in stocks (inflows/outflows); stocks accumulate over time; feedback loops drive growth or stability (positive loops amplify exponentially, and negative loops promote equilibrium); multivariable systems are represented by state-space models (nonlinearities like predator-prey dynamics can be used here, e.g., oxygen delivery in ICU); delays model biological lags (Euler or Runge-Kutta methods numerically solve them); auxiliary variables and sensitivity analysis add flexibility for scenario testing of medical interventions.

System dynamics on the circulatory system show how it is a complex adaptive network that regulates homeostasis by feedback. Stocks are blood volumes in compartments (e.g., ventricles or aorta), flows transfer, and loops respond—positive amplification during exercise and negative pressure stabilization. Equations are blood volume changes, cardiac output ($HR \times SV$), and BP ($CO \times PR$). Baroreceptor reflexes model nonlinear adjustments. Simulations with differential equations show how perturbations (MI) disrupt loops, lowering output and triggering compensations.

Combining these models with real-time data makes them clinically useful in the ICU: wearables/sensors/health records update parameters dynamically via machine learning (Bayesian methods). Cloud computing infrastructure and standardized APIs are essential for ensuring interoperability. Dashboards show outputs. Alerts predict adverse events. Personalized adjustments optimize therapies (fluid administration). Data privacy validation of models User training not to over-rely.

We developed the model for simulation in NetLogo with six compartments for heart chambers and vessels and over 30 parameters (volumes, elasticities, and resistances) governed by equations relating volume-pressure-flow (Figure 3.1). Ventricular pressure has time-varying elastance and curvature factors as well. Normal initial conditions are simulated on sliders, outputs (e.g., cardiac output) are monitored, and dynamics are plotted (Figure 3.2). Elastance can be changed to simulate myocardial infarction (linking digital twins to personalized monitoring).

For myocardial infarction simulations, time series of pressures, outputs, and volumes match clinical data: pressure-volume loops shrink (ventricular function degraded) (Figure 3.3). Kolmogorov-Chaitin complexity is used for advanced analysis: drops post infarction show loss of dynamic range (Figure 3.4). These metrics (along with spectral or chaos indicators) provide novel means of monitoring circulatory stability in intensive care.

This chapter goes from simpler three-compartment models to granular ones with more controls to generate scalable data for long-term studies. Future applications include real-time data acquisition, processing algorithms, decision support tools, and privacy protection. System

dynamics explains mechanisms of myocardial infarction; personalized intensive care is possible with predictions of responses and interventions (complexity and validation still need to be addressed, but AI synergies will). Multisystem models and clinical trials will expand the potential.

In summary, this chapter shows system dynamics modelling as a tool for circulatory complexity in intensive care theory to bridge the gap in simulation. For future research, validate models against real patient data, multisystem interactions like respiratory coupling, AI-enhanced real-time platforms, and clinical trials to assess impact on outcomes. These steps will refine intelligent methods to provide proactive systems informed by critical care strategies.

4. HYBRID MODELING IN INTENSIVE CARE: COMBINING SYSTEM DYNAMICS AND AGENT-BASED MODELING TO IMPROVE PHYSIOLOGICAL SIMULATIONS

4.1. Hybrid Modeling in Medicine and Intensive Care

Computational modeling techniques can transform medicine (especially when simulations of physiological processes and patient outcomes are needed). Hybrid modeling refers to a technique that combines system dynamics modeling (SDM) with agent-based modeling (ABM). SDM models the macro-level behavior of systems (feedback loops and differential equations), while ABM models agents that interact based on rules (emergent behaviors mimic heterogeneity from the real world, such as cellular interactions or individual patient responses in clinical settings).

In medicine, hybrid SDM-ABM models are being used to overcome the shortcomings of single-method approaches (SDM alone may oversimplify emergent phenomena; ABM becomes computationally expensive without structured macro-level constraints). The hybrid paradigm combines both: SDM provides a high-level framework for continuous processes and ABM granularity for discrete events and spatial dynamics. This approach is important in ICM, where real-time decision-making involves multiple physiological systems, medical interventions, and environmental factors; for example, ICM hybrid models can simulate patient trajectories under different conditions (sepsis progression or ventilator management) to enable predictive analytics for personalized care.

Hybrid modeling started as early as the 2000s in public health and epidemiology studies [95]. Epstein's seminal work on agent-based models of infectious diseases laid the foundation, but system dynamics added broad application to chronic and acute care too. In intensive care, hybrid models have been used to simulate organ system failures (SDM fluid dynamics and metabolic rates; ABM cellular responses to inflammation/infection). A benefit is multiscale modeling: from molecular agent interactions to organ-system feedback loops. This multiscale integration leads to more accurate predictions in dynamic systems (e.g., ICU) where patient states change rapidly due to interventions or complications.

One of the main advantages of hybrid SDM-ABM models in medicine is their predictive power: continuous (SDM) and discrete (ABM) components can forecast outcomes more accurately; e.g., in cardiovascular intensive care, SDM could model blood pressure regulation by differential equations while ABM would simulate endothelial cell behaviors that influence vascular tone, so better simulations of shock states may be developed [96]. Another advantage is

visualization/interpretability: ABM components often have spatial representations, making physiological processes more accessible to clinicians and educators. In ICM educational settings hybrid models are used as interactive tools for trainees to experiment with scenarios such as acute respiratory distress syndrome (ARDS), where the SDM tracks oxygenation while the agent-based model (ABM) simulates alveolar recruitment.

Challenges in implementing hybrid models are worth mentioning. Synchronizing SDM continuous time steps with ABM discrete events is computationally expensive (often on advanced software platforms like AnyLogic or NetLogo extensions). Validation is another challenge; hybrid models require data sources from clinical trials to real-time monitoring (which can be hard to obtain in ICM due to ethical restrictions) [97]. Hybrids increase parameter uncertainty because aggregate data are used for SDM and individual rules for ABM (inconsistency possible); sensitivity analysis is needed to identify influential parameters but takes longer to develop. Integration challenges are interfacing the two paradigms: stock-flow diagrams of SDM must match agent rules of ABM (custom algorithms for data exchange often required) [98].

In the ICU specifically, hybrid modeling fills an under-explored area of real-time physiological monitoring. Models in ICM are usually isolated systems, but hybrids allow integration (SDM for hemodynamic stability and ABM for immune cell dynamics in sepsis, e.g.). Hunter et al. studied this problem using a hybrid model of hemodynamics in intensive care units [99]. Hybrids showed these benefits in modeling multi-organ dysfunction (SDM captured systemic inflammation loops; ABM simulated cytokine release by individual cells), improving outcome predictions in simulated cohorts [99]. Benefits extend to resource allocation: pandemic ICU hybrids have modeled bed occupancy (SDM) and patient contagion risk (ABM) for triage [26, 99], but data integration remains an issue: EHRs provide macro-data for SDM, but micro-level details are missing for ABM, so imputation bias is introduced.

The hybrid model also facilitates scenario testing, which is a critical ICM benefit because randomized trials are rarely feasible. Maternal-fetal intensive care systems depend on each other, so hybrid models can simulate how oxygen moves between them, including SDM bulk flows and ABM placental exchanges. This applies to emerging applications in obstetric anesthesia monitoring fetal heart rate variability, where timely interventions prevent hypoxia. Scalability is an issue here (as the number of agents increases, simulation time goes up, and it is too demanding for HPC) [100]. Validation is ethical, as fetuses are invasive to collect data from. This forces researchers to use animal models or historical data, which may not work perfectly for humans.

But the benefits outweigh the challenges when used judiciously: hybrid models foster interdisciplinary collaboration between biomedical engineers, clinicians, and data scientists; in

education they allow “what-if” analysis (changing vasopressor doses in shock simulations); and ML integration for adaptive tuning of parameters will help overcome uncertainty challenges in future directions.

So, hybrid SDM-ABM modeling is a versatile tool for medicine (intensive care especially) because it can handle complexity: benefits are predictive accuracy, visualization, and scenario exploration; challenges are computational demands, validation, and integration. As we move to examples [101] like maternal-placental-fetal interactions, this background tells us how pioneering it is to apply hybrids to underexplored areas (fetal oxygen delivery in ICUs).

4.2. Model framework: Combining SDM and ABM for Simulating Maternal-Placental-Fetal Interactions

We move from the general; this model builds on previous hybrid applications in healthcare (e.g., Djanatliev et al. [103] for health systems research, such as emergency department flows) and the principles of hybrid modeling introduced in the background to a specific example of a hybrid SDM-ABM model for maternal-placental-fetal interactions (in particular simulating and estimating fetal oxygen delivery) in this section. _This is the first attempt in intensive care medicine (OB/neonatology) where direct monitoring of fetal physiological parameters is limited or even invasive. SDM for macro-level physiological simulations based on aggregate flows and feedback loops and ABM for micro-level visualization of individual interactions (e.g., oxygen diffusion across the placental barrier) fills two major gaps in real-time fetal health assessment.

The reason for the hybrid framework is that the maternal-placental-fetal unit is complex: hemodynamics (blood pressure, cardiac output) can be measured noninvasively and in real time by SDM components, but fetal conditions (especially oxygen status) are inferred indirectly from methods such as fetal heart rate monitoring or Doppler ultrasound, which are not precise enough for predictive interventions. The placenta is an intermediary organ for nutrient and oxygen exchange, so it has continuous (bulk blood flow) and discrete (oxygen movement) behaviors, making it a good candidate for hybrid modeling. SDM models the continuous parts (ODEs for volume changes, pressures, and oxygen fluxes), while ABM introduces stochastic elements to model heterogeneous agent behaviors (e.g., variable diffusion rates depending on local placental architecture).

In practice, this approach means that simulation software supporting hybrid paradigms is used (NetLogo for visualization of ABM coupled with SDM solvers); the user interface for clinical and educational use is interactive so parameters can be manipulated and output observed in real time: a scenario selector for predefined cases (normal pregnancy vs. maternal anemia), graphical

plots for time-series data on maternal and fetal circulatory parameters, numerical monitors showing critical values like cardiac output (CO), and FD_{O_2} visual windows showing placenta and fetal oxygen delivery processes [102].

Figure 4.1 shows the model interface in a “Normal Test” scenario. The top-left scenario chooser lets us switch between different simulation modes quickly. Below it are two plotting areas showing temporal variation of the circulatory systems: the upper plot shows maternal (e.g., aortic pressure and venous volume), whereas the lower plot shows fetal equivalents; central monitors give numerical readouts during simulations so quick assessments can be made while running simulations. The central visualization window uses ABM to represent the placenta as a spatial grid where agents (red blood cells and oxygen) interact; on the right, another window dynamically shows oxygen delivery to the fetus (symbolic for transfer efficiency). Bottom sliders allow fine-tuning parameters (the left side controls maternal/placental variables (hemoglobin concentration, placental resistance), and the right side fetal aspects (heart rate, blood volume)).

The interface is not only easier to use but also fits the goal of intensive care training, where visual feedback helps students understand abstract ideas. For example, in a clinical simulation, sliding the slider to induce maternal anemia decreases FD_{O_2} in the SDM plots.

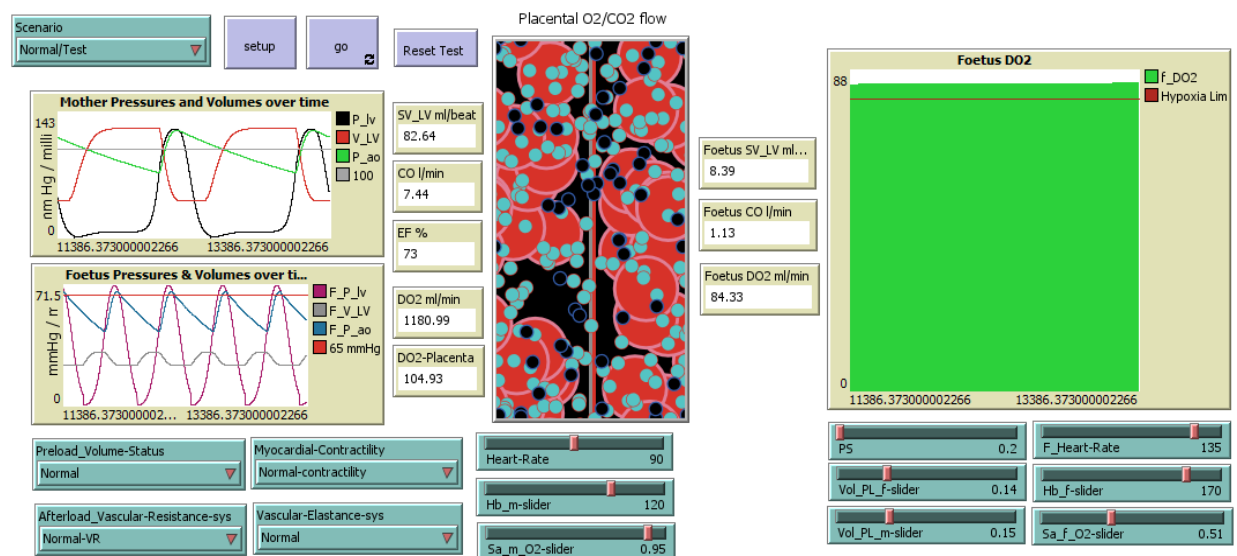


Figure 4.1. A complete view of the prototype interface. The scenario selector (upper left corner) says "Normal." Central monitors display numerical values of parameters. On the left there are plotting areas for maternal (upper graph) and fetal (lower graph) circulatory dynamics over time; the placenta is shown in the central pane and oxygen supply to the fetus in the right pane; model parameters can be changed with lower sliders. [102]

The underlying SDM structure (Figure 4.2) partitions the model into three subsystems: (1) maternal circulatory system, (2) fetal circulatory System, and (3) placenta as exchange hub. Squares are stocks (e.g., blood volumes in compartments), and diamonds are auxiliary variables (e.g., elastance or resistance). V stands for volume, P for pressure, E for elastance, and R for resistance; subscripts lv for left ventricle, ao aorta, sys systemic circulation, etc., and fetal and placental elements (e.g., pl placenta).

The ABM is integrated into this SDM framework at the placental level, where the scenario selected sets rules for agents (SDM-calculated maternal blood flow in a certain scenario sets rules for influx of agents into the placental window, e.g., probabilistic movement of oxygen from the maternal site of the placenta to the fetal site).

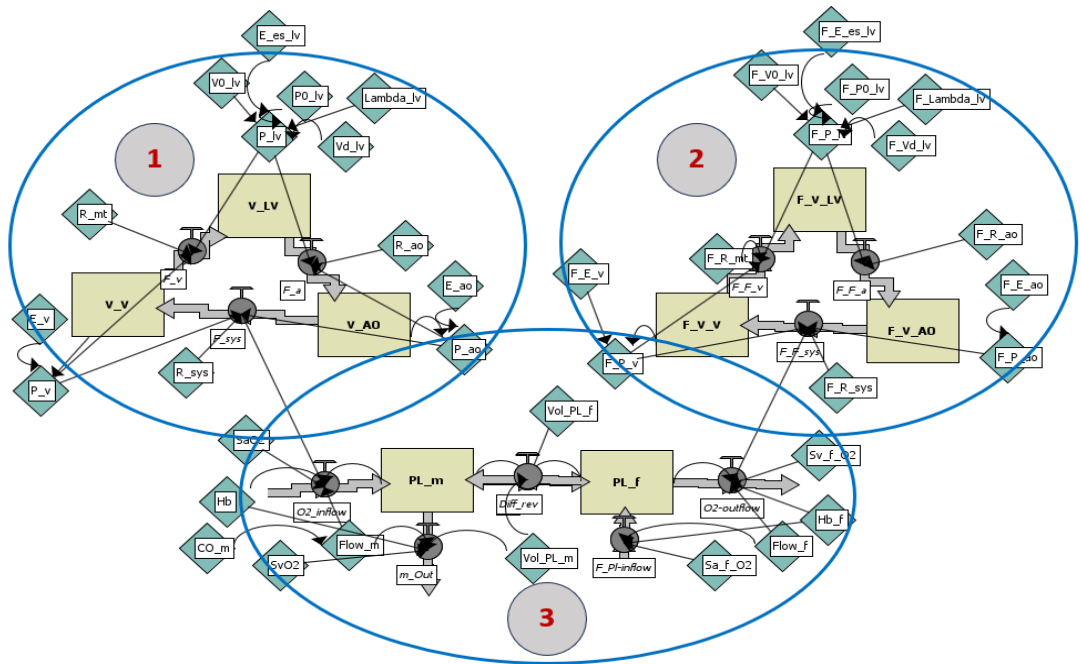


Figure 4.2. The model's system dynamic scheme: (1) the circulatory systems of the mother, (2) the fetus, and (3) the placenta; squares represent stocks and diamonds other characteristics. A particular parameter is indicated by capital letters (V for volume, E for elastance, etc.). [102]

In general, this model builds on previous hybrid applications in healthcare (e.g., Djanatliev et al. [103] for health systems research like emergency department flows where SDM was used for queues of patients and ABM for individual behaviors). In IC we are similar to works like [103-105] that used loosely coupled SDM-ABM for prospective health technology assessments: simulating population dynamics (SDM) with individual patient variabilities (ABM). But our focus on physiology rather than operations is an understudied area (especially fetal medicine, where hybrid models are rare).

Benefits in intensive care settings are multifaceted: clinically it is a decision support tool (predicting fetal distress from maternal inputs); timely intervention (supplemental oxygen or emergency cesarean sections) can be done based on model outputs; educationally it allows medical students to see cause-effect relationships (e.g., how maternal anemia propagates through placental exchange to fetal hypoxia). Similar challenges exist in the background: calibration needs different data sources (ultrasound for placental parameters, pulse oximetry for maternal oxygen saturation); SDM deterministic outputs and ABM variability may differ.

In summary, this hybrid framework illustrates how SDM-ABM can be applied to maternal-placental-fetal dynamics (i.e., a tool for simulation in intensive care). It combines macro-level physiological modeling with micro-visualization to fill monitoring gaps; it allows personalized obstetric care to emerge.

4.3. Mathematical and Computational Foundations of the Hybrid Model

The math behind the hybrid model is lumped parameter approximations of cardiovascular physiology, which are then used to model the maternal-placental-fetal triad (SDM for ODEs and ABM for rule-base algorithms). This part describes equations in great detail, how they were derived and integrated, and how they make it possible to simulate delivery of oxygen to the fetus as the main outcome variable.

The circulation is modeled by ODEs describing volume changes in circulatory compartments (fluid dynamics and elasticity). The cardiovascular system is modelled as a network of compliant vessels and resistive elements, just like electrical circuits where volume corresponds to charge, pressure voltage, and flow current. For the maternal system, the left ventricular volume change is modeled as:

$$\frac{dV_{lv}}{dt} = \frac{(P_v - P_{lv}) \times k}{R_{mt}} - \frac{(P_{lv} - P_{ao}) \times k}{R_{ao}}. \quad (4.1)$$

This equation balances inflow through the mitral valve (first term) against outflow through the aortic valve (second term), where k is extracardiac regulatory factors such as autonomic influences.

Similarly, arterial (aortic) volume dynamics are:

$$\frac{dV_{ao}}{dt} = \frac{(P_{lv} - P_{ao}) \times k}{R_{ao}} - \frac{(P_{ao} - P_v) \times k}{R_{sys}}. \quad (4.2)$$

And venous volume:

$$\frac{dV_v}{dt} = \frac{(P_{ao} - P_v)}{R_{sys}} - \frac{(P_v - P_{lv})}{R_{mt}}. \quad (4.3)$$

Pressures are modeled from elastance-volume relationships (vessel compliance):

$$P_{ao} = E_{ao} \times V_{ao}, \quad (4.4)$$

$$P_v = E_v \times V_v. \quad (4.5)$$

For the left ventricle, a time-varying elastance model describes systolic contraction:

$$P_{lv} = tve \times E_{es_{lv}} \times (V_{lv} - V_{d_{lv}}) + (1 - tve) \times P0_{lv} \times (\exp \exp (\lambda_{lv}(V_{lv} - V0_{lv})) - 1), \quad (4.6)$$

where tve is a periodic function simulating the cardiac cycle, E is the maximum elastance, $V0$ is unstressed volume, and $P0$ is a diastolic baseline.

These equations extend to the fetal system with parallel structures (lower resistances, higher heart rates, etc.). The placenta links them together via compartments that model intervillous spaces. The placenta is modeled using ODEs behind a two-compartment Michaelis-Menten transporter. Parameters in this submodule can also be adjusted using sliders on the user interface.

Oxygen delivery (DO_2 or DO_2) calculation combines this hemodynamics with oxygenation parameters:

$$DO_2 = CO \times 1.34 \times Hb \times S_aO_2, \quad (4.7)$$

where CO is cardiac output (L/min), Hb is hemoglobin (g/L), and S_aO_2 is arterial oxygen saturation (via oxyhemoglobin dissociation curve), and the constants account for bound oxygen.

The ABM layer connects to these SDM outputs by discretizing flows into populations of agents: each agent represents a cohort of erythrocytes (attributes like oxygen level), and movement rules are probabilistic (agents move according to scenario type). Computationally the model uses numerical integration (e.g., Runge-Kutta for ODEs) with time steps synchronized to ABM update.

It builds on models in critical care (e.g., for ICU patient flows by Dong et al. [105]) but extends it to fetal oxygen; assumptions of incompressibility and neglect of inertial terms are valid for low-frequency dynamics but may not be correct for rapid changes (contractions). Fuzzy uncertainties for parameter variability could be added later [106].

4.4. Scenario-based Simulations: Pathophysiological Dynamics

To show how the model can be used practically, this section considers a range of simulation scenarios, each designed to represent typical intensive care situations in obstetrics (the hybrid framework is used to predict FD_{O_2} responses, and information about interventions is available from them). Plots and ABM windows display outputs, and results are compared with clinical literature.

The Normal scenario gives us the baseline behavior. If we set all parameters to physiological values (maternal Hb 120 g/L, heart rate 80 bpm, placental resistance low), then model output FD_{O_2} , is stable around 150 ml/min, which is typical for healthy term pregnancies. Increasing maternal heart rate (to 100 bpm) increases maternal CO and therefore FD_{O_2} by 20% (compensation). A mild decrease in hemoglobin (to 100 g/L) decreases FD_{O_2} proportionally (sensitivity). In figure 4.3(a), oxygen delivery as a green area plot over time remains above the red hypoxia threshold (80 ml/min), and the monitors show steady-state values. .

In the Maternal Anemia Scenario, hemoglobin progressively drops from 110 g/L to 90 g/L at 1000 simulation ticks (arbitrary units of time), resulting in a 15% FD_{O_2} decrease; it then further declines to 70 g/L at 1300 ticks, which increases the drop of FD_{O_2} to 30%. Additionally, the introduction of fetal anemia (reducing fetal hemoglobin to 10 g/dL) at 1500 ticks brings the level below the threshold. ABM visualization shows fewer oxygenated agents reaching the fetal side. Figure 4.3(b) shows this decline (the green area shrinks, crossing the red line alerting us). This finding is consistent with studies on anemia in pregnancy [107], where FD_{O_2} reductions correlate with growth restriction and influence transfusion thresholds.

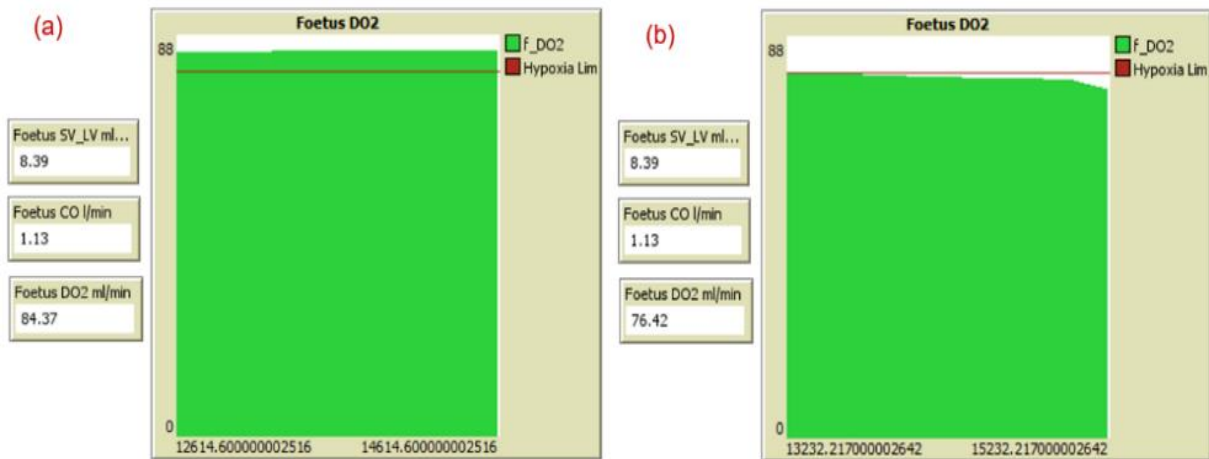


Figure 4.3. The green area and the lower display show the oxygen supply to the fetus (standard conditions a) and maternal iron deficiency (b). The threshold for fetal hypoxia is shown by the red line. [102]

The Abruptio Placentae Scenario represents acute detachment (emergency case). At 1200 ticks, partial abruption halves exchange; full simulation at 1300 ticks further reduces it. The exchange rate drops to <50 ml/min, the central ABM window darkens red (ischemia), and the simulation stops at 1700 ticks (fetal death). Fig. 4.4(a) shows the drop: model is able to capture time-sensitive pathologies. Compared to a real case, this mimics the Doppler finding of reversed flow, so training for rapid diagnosis [100] can be done here.

The Uterine Contractions Scenario adds periodic pressure spikes (every 200 ticks), as in labor. Each contraction causes a FD_{O_2} dip of 10-15% but doesn't breach hypoxia limits in normals but risks it in compromised states (see Fig 4.4(b)). Green area oscillates red in intrapartum hypoxia risk monitoring.

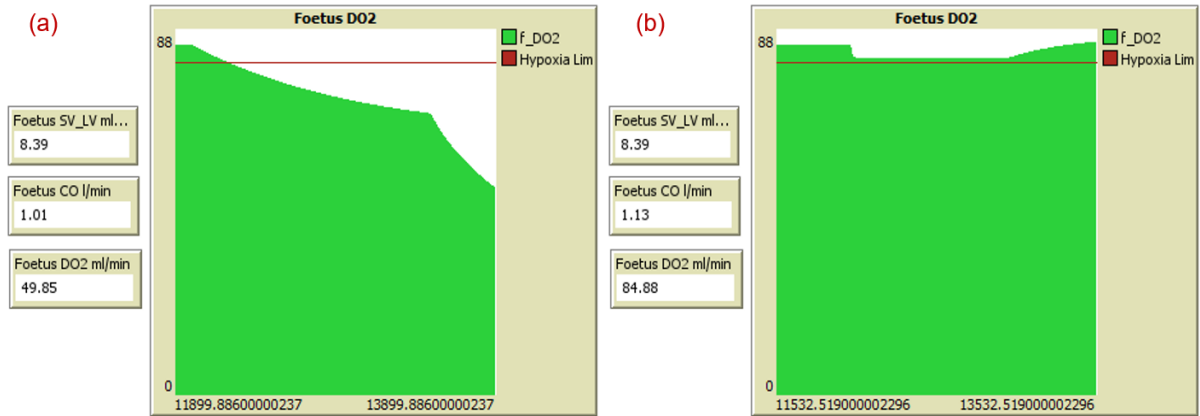


Figure 4.4. The fetus oxygen delivery (lower monitor and green area) in (a) Abruptio Placentae scenario and (b) typical uterine contraction. The fetal hypoxia limit is shown by the red line. [102]

Figure 4.5 shows the picture for complex pathology (e.g., heart failure with maternal anemia) and the general anesthesia effect.

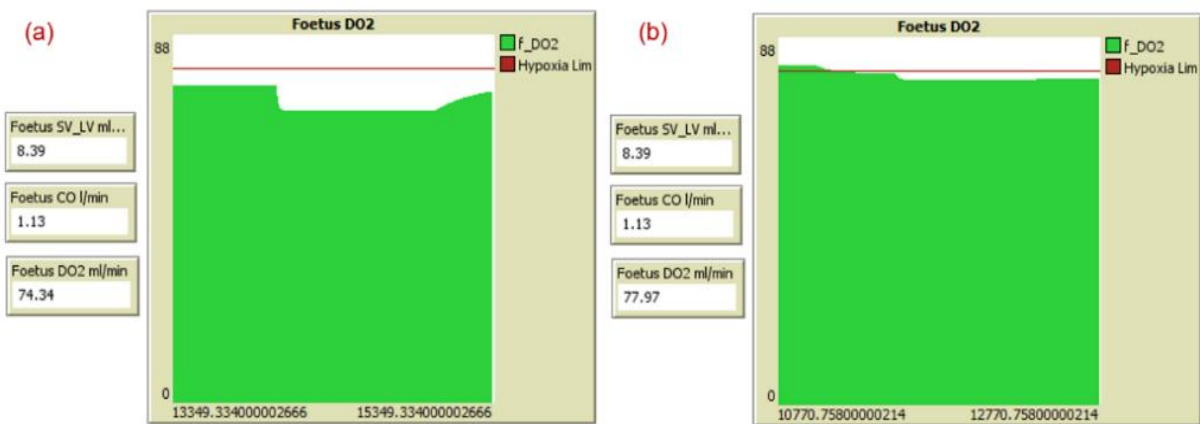


Figure 4.5. Green region and lower monitor show oxygen delivery to the fetus in (a) complex pathology and (b) anesthesia. The red line shows the fetal hypoxia limit. [102]

Results are compared to benchmarks. In intensive care these simulations help plan scenarios (e.g., contraction tolerance in anemic mothers). “What if?” exploration is educational too; deeper pathophysiological understanding results.

4.5. Applications in Clinical Practice, Educational Contexts, Limitations, and Future Development

The applications of this hybrid model are clinical (decision support systems), educational (teaching), and research (intensive care). Clinically it is a prototype for DSSs: it simulates patient-specific scenarios to forecast and guide interventions (for anemia modeling, hemoglobin thresholds inform transfusion; for abruption, deterioration time can be predicted for timing of surgery, etc.). EHR integration would allow real-time inputs to fill fetal data gaps [108].

In education, interactive features allow you to explore parameter effects (e.g., preload on) so that complex interactions can be understood better; in ICM training, it simulates rare events safely.

Limitations include simplifications: lumped parameters ignore gradients; no ventilation modelling affects accuracy in respiratory distress; and valve assumptions neglect regurgitation. Fixed parameters ignore adaptation; inertial omissions fit steady but not pulsatile flows [46].

Future directions are clinical validation, intravascular details, and machine learning for personalization. KCC estimation [109] from accessible parameters (maternal hemodynamics, fetal heart rate) matches in Figure 4.6 for abruption (blue KCC, red), suggesting noninvasive monitoring.

It would be important to use measures available in a clinical setting to estimate the amount of oxygen given to the fetus. Using KCC estimation based on three maternal parameters (three from the mother's side and one from the fetus, i.e., fetal heart rate), researchers found that KCC values represent a relationship that needs further investigation (i.e., might be a prospective research direction).

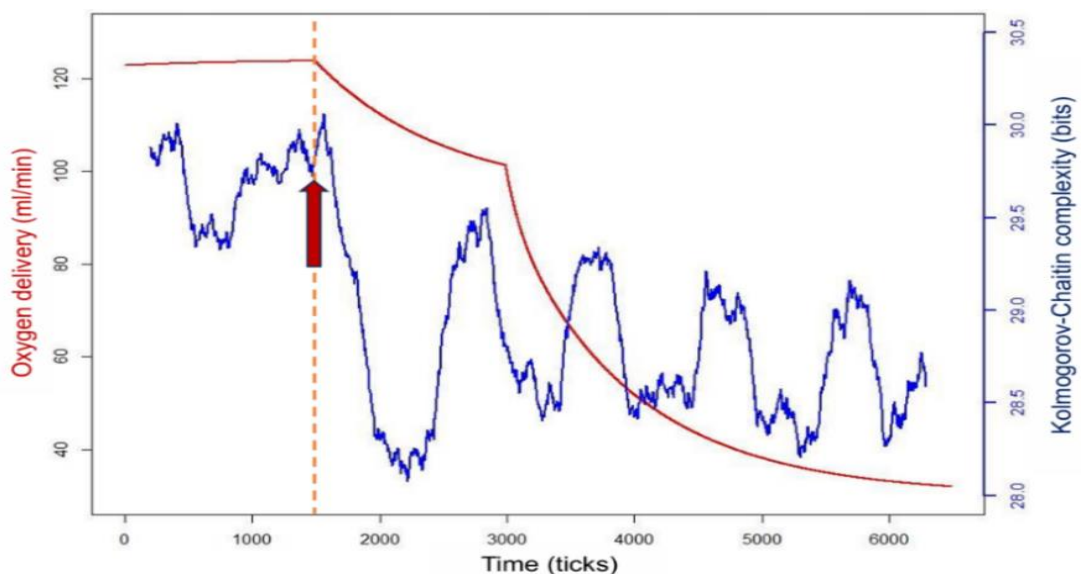


Figure 4.6. Oxygen delivery to the fetus in abruptio placentae (red color line) and Kolmogorov-Chaitin complexity (blue color line).

4.6. Digital Twins in ICM

Digital twins have been used in aerospace and manufacturing for years to create virtual replicas of physical assets for real-time monitoring and predictive maintenance but are now spreading into healthcare, where they can be used to revolutionize ICM: A digital twin is a dynamic, data-driven virtual model of a patient or physiological system that mirrors its real-world counterpart, continuously updated with sensor data, EHRs, etc., to simulate states, predict outcomes, and optimize interventions (hybrid SDM/ABM models were introduced earlier in this chapter, where SDM models macro-level continuous processes such as hemodynamics flows, while ABM models micro-level discrete interactions such as cellular exchanges). By extending these hybrid models to digital twins, ICM can achieve unprecedented personalization, especially when maternal-placenta-fetal interactions are complex (direct fetal monitoring not possible).

In IC settings digital twins fill the need for holistic real-time representations of patients in high-stakes, rapidly changing conditions. ICM (traditional ICM models or isolated physiological monitors) only provides static models/monitors, but digital twins combine multimodal data (vital signs, imaging, lab results, and even genomic information) to create a virtual patient. In acute hypoxemic respiratory failure digital twins have been used to model lung mechanics and predict how patients will respond to NIV (this shows that adding variability based on patient is more like what happens in real life [110]). This is similar to how the hybrid SDM-ABM model can simulate fetal oxygen delivery, where SDM equations for maternal cardiac output and placental resistance could be fed with real-time data from maternal monitors while ABM visualizes emergent behavior (e.g. diffusion of oxygen across placenta). Predictive analytics can forecast fetal distress before it manifests clinically so interventions (e.g. maternal oxygen supplementation or position change) can be made proactively.

Digital twins are used in ICM not only for simulation but also decision support and automation. Critical care is a domain where resources are limited and mistakes by people can change the outcome so digital twins can automate scenario testing (just as the chapter describes how pathophysiological dynamics work). A digital twin framework for discrete event simulation in ICU workflows optimizes bed allocation and staffing by modelling patient trajectories with real-time state changes [111]. In the obstetric ICU, a digital twin of the maternal placenta-fetal unit could personalize care during high-risk pregnancies using inputs like fetal heart rate variability and maternal blood pressure to estimate dynamically FD_{O_2} . This goes back to the interface of the hybrid model sliders for parameters like hemoglobin concentration, which could be replaced with automated data streams from wearables or EHRs so clinicians can visualize “what if” scenarios (e.g., effects of anesthesia on placental perfusion) in real time.

The benefits of digital twins in ICM are multiple: they reinforce hybrid modeling advantages but scale them up (data assimilation). Calibration continuously improves prediction accuracy (could lower death rate in sepsis or multi-organ failure by giving early warning signs), as seen in scoping reviews of healthcare digital twins [80]. Visualization is still a strength; ABM components in a digital twin can render spatial dynamics so abstract processes like cytokine storms in maternal inflammation become understandable to interdisciplinary teams. Digital twins are training tools too—trainees experience rare events (abruptio placentae) without putting patients at risk and understand the material better [112]. In research they enable virtual trials—ethical barriers in fetal medicine are overcome by testing interventions on twin cohorts from historical data.

Nevertheless, deployment challenges are similar to hybrid models (computational complexity and validation). High-fidelity simulations need real-time data synchronization (edge computing for ABM stochastic elements without latency so decisions in ICU don't become delayed); EHRs only provide macro-level inputs for SDM, whereas micro-level details (e.g., cellular heterogeneity) of ABM require advanced sensors or imputation so bias or inaccuracy may occur; digital twins process health data, which raises privacy and ethical issues, and regulations like HIPAA apply; validation against different patient populations is needed, but ICM studies are rare because they are heterogeneous; studies have shown prospective trials are needed to confirm twin fidelity [113]. Uncertainty in parameters from real-time updates needs sensitivity analysis and machine learning for adaptive tuning (see previous sections).

Digital twins are a natural extension of the hybrid SDM-ABM paradigm for intelligent ICM systems. AI could automate parameter estimation (e.g., fetal oxygen simulations) to remove uncertainties. In maternal-fetal care, twins could proxy (Kolmogorov-Chaitin complexity (KCC) approximations in abruption scenarios correlated with fluctuations). Consortium standardization is one possibility beyond isolated ICU applications (pandemic response or chronic care transitions). Digital twins are the complexity science approach of this thesis at macro and micro scales for resilient patient-centered intensive care.

4.7. Synthesis of Chapter 4

In this chapter we discuss SDM and ABM as a hybrid method for modeling complex physiological processes in ICM (mother-placental-fetus interactions estimating fetal oxygen delivery) and also emerging concepts like digital twins. The hybrid method takes advantage of SDM's ability to model macro-level continuous processes (feedback loops, differential equations for aggregate flows) and ABM's micro-level discrete interactions between autonomous agents

(emergent behaviors, spatial dynamics). SDM alone may not model individual heterogeneity, while ABM is computationally expensive without structure.

In ICM hybrid modeling improves simulations of patient outcomes by modeling multiscale phenomena (cellular responses to sepsis, organ-system interactions in ventilator management), SDM models continuous elements (fluid dynamics, metabolic rates) ABM discrete events (immune cell behaviors or inflammation spread). Predictive accuracy, visualization and scenario testing are improved. ICU is a dynamic environment where changes due to interventions or complications happen rapidly so decision making has to be done in real time. Better prediction of shock states or organ failures; protocol development; personalized care can be achieved. Visualization with ABM makes abstract processes accessible for education and training (simulate acute respiratory distress syndrome: SDM tracks oxygenation, ABM alveoli interaction).

However, problems arise (computational complexity due to synchronizing continuous and discrete elements; validation is hard in ethically sensitive ICM settings with scarce data; parameter uncertainty increased by dual paradigms). Integration means SDM stock-flow diagrams have to be aligned with ABM agent rules (often software needed). In ICM hybrids are used for comprehensive simulations (hemodynamic stability with immune dynamics in sepsis or resource allocation during pandemics by simulating bed occupancy along with contagion risk). Scenario testing is particularly useful when trials cannot be done (e.g. maternal-fetal care for how oxygen moves between the two).

The chapter then describes a specific hybrid model framework for modelling maternal-placental-fetal interactions with special emphasis on fetal oxygen delivery (i.e., FD_{O_2}). This model fills gaps in real time fetal monitoring: maternal hemodynamics are easily measured but fetal status is inferred indirectly. SDM solves ordinary differential equations (ODEs) for macro level continuous flows (blood pressure and cardiac output), ABM shows micro level discrete events (red blood cell movement and diffusion of oxygen across the placental barrier). In software such as NetLogo, it has scenario selectors, graphical plots for maternal and fetal circulatory dynamics, numerical monitors for cardiac output and FD_{O_2} , visual windows displaying placenta as grid spatial representation of agents colored by oxygenation levels, sliders to change parameters divided into maternal-placental and fetal variables so that conditions like maternal anemia can be simulated and immediately observed effects on agent flow and FD_{O_2} (Figure 1).

The SDM structure splits into three subsystems: maternal circulation, fetal circulation and placenta as exchange hub. Stocks are blood volumes and auxiliaries for pressures, elastances and resistances (Figure 4.2). ABM couples at the placental level where SDM flows feed influx of

agents. Loose coupling means SDM does continuous computations while ABM provides stochastic granularity for visualization. Unlike previous hybrids in healthcare which typically model operational flows (e.g., emergency departments), this model simulates physiological processes for fetal medicine so it can be used as a decision support tool to predict distress or an educational platform to study cause-effect relationships.

Mathematically, the model uses lumped-parameter ODEs for circulatory dynamics (systems are compliant vessels with resistive elements). Equations relate volume changes in compartments (e.g., left ventricle) to inflows and outflows plus regulatory factors; pressures come from elastance–volume relationships (elastance time varies during cardiac cycles); these also apply to fetal and placental compartments, where FD_{O_2} is calculated from CO , hemoglobin, and oxygen saturation. ABM discretizes flows into agents (e.g., oxygen load) and probabilistic rules based on SDM gradients for movement/exchange of such flows. Numerical integration synchronizes time steps, and sensitivity analysis identifies key influencers. Fluid incompressibility is an assumption but adequate for low-frequency dynamics but not rapid changes.

Scenario-based simulations show how the model is applicable in obstetric ICM (Figures 4.3 to 4.5). Normal scenario gives a steady rate of FD_{O_2} around 150 ml/min, but adjustments are made (model shows compensatory mechanisms e.g., heart rate increases blood delivery); maternal anemia decreases FD_{O_2} progressively crossing hypoxia limits when combined with fetal anemia (less oxygenated agents); abruptio placentae simulates acute detachment which drops FD_{O_2} sharply to critical levels (ischemia); uterine contractions cause periodic FD_{O_2} dips tolerable in normals but dangerous in compromised states; more complex pathologies (heart failure + anemia), anesthesia effects further decline plots show fluctuations against hypoxia limits. These match clinical observations and help plan intervention and training.

Clinical practice uses the model to forecast FD_{O_2} (to guide transfusions or surgery) and education to safely explore rare events. Some limitations are the exclusion of spatial gradients and ventilation modeling, the constancy of fixed parameters over time, the potential inaccuracy of pulsatile flows, etc. Validation is possible, and there are several intriguing research directions, including adding details about intervascular exchanges, using machine learning for personalization, and exploring non-invasive monitoring through proxies like Kolmogorov-Chaitin complexity approximations (KCC) that correlate with abruption cases (Figure 4.6).

The chapter ends with digital twins as an evolution of hybrid modeling in ICM. Digital twins are virtual patient replicas updated from sensors and EHRs in real time; SDM for macro-processes and ABM for micro-interactions. In maternal-fetal care they enable dynamic FD_{O_2} estimation

(predict distress), scenario testing (optimize interventions), and visualization (for resource allocation). The benefits of digital twins are similar to those of hybrids, including accuracy, visualization, and efficiency; however, they also present challenges such as computational load, data integration, privacy, and validation. Additionally, AI integration could automate tuning and standardize applications to ensure resilient patient-centered care.

Hybrid SDM-ABM modeling at the end of this chapter shows how ICM can be advanced through multiscale simulations (fetal oxygen delivery as an example). It fills gaps in monitoring and decision-making (digital twins extend it to personalized intelligent healthcare). Challenges notwithstanding, this approach promotes interdisciplinary innovation (complexity science perspective) for outcomes improvement in high-stakes situations.

5. FROM TRADITIONAL MEDICINE TO COMPUTATIONAL PARADIGMS: COMPLEXITY-DRIVEN APPROACHES TO DIAGNOSIS AND PROGNOSIS IN CRITICAL CONDITIONS

5.1. Medical diagnosis and prognosis: tracing the evolution from ancient times to the present era

5.1.1. Historical perspective on the concept evolution

The word “diagnosis” has a range of definitions in the Merriam-Webster dictionary [114]:

(a) recognition of a disease from its signs and symptoms; (b) a determination reached through the diagnostic process; (c) an inquiry or analysis into the cause or nature of a condition, situation, or problem; (d) a statement or conclusion resulting from such an analysis; (e) in biology, a concise technical description of a taxon. In this chapter, we restrict the use of “diagnosis” to its medical sense.

This work spans multiple domains (notably computer science — machine learning and artificial intelligence — and intensive care medicine), and it is particularly relevant to primary stakeholders (patients and clinicians). In the following section we treat both diagnosis and prognosis (the temporal aspect of diagnosis). Table 5.1, based on sources [115–118], consolidates the principal milestones and innovations in fields pertinent to this research. After briefly outlining the key developments referenced in the table, further details are provided below.

Findings and developments connected with the present research indicate that medical practice has been governed by behavioral codes for millennia (e.g., the Code of Hammurabi, ca. 1740 BC, and the Hippocratic Oath circa 460–360 BC), with disease understanding shaped by political and religious constraints [119]. Nevertheless, formal mathematical decision-making in medicine lacked foundations in symbolic logic and probability until the emergence of modern decision theory in the 1950s [120, 121]. Prior to the early 1970s, diagnostic research predominantly relied on statistical methodologies [122]. Later, AI approaches grounded in knowledge representation and logic led to a decline in the appeal of purely statistical methods. The 1980s witnessed a swift expansion of logic-based “expert systems,” which yielded highly successful clinical decision-support systems (CDSS) [123, 124].

Table 5.1. Discoveries/developments related to the current research

Date	Discoveries/developments
c. 400 BC	The Greek physician Hippocrates introduces the concept of disease.
...	
c. 1660	Dutch naturalist Antonie van Leeuwenhoek develops the light microscope.
...	
1810	French physician Rene' Laennec invents the stethoscope.

Table 5.1. Continued

1900	The widespread use of chest X-rays enables early detection of tuberculosis, the leading cause of death at the time.
...	
1906	Dutch physiologist Willem Einthoven invents the electrocardiograph (ECG) to monitor and record the electrical activity of the heart.
...	
1936	Alan Turing presents the concept of a universal machine, later known as the Turing machine.
...	...
1956	The Dartmouth Summer Research Project on Artificial Intelligence takes place.
1980s	Rule-based expert systems emerge.
...	...
2000s	The concept of artificial intelligence experiences a resurgence.
2010	Deep learning becomes prominent.
2016	The first reprogrammable quantum computer is developed.
2018-present	The USA Food and Drug Administration establishes a regulatory framework for Software as Medical Device (SaMD), followed by similar actions by the European Union.

During the 1990s and 2000s institutional and research interest in AI and automated medical decision-making declined, as did work on expert systems (with some exceptions [125]). The Internet played a pivotal role in disseminating medical knowledge, and information systems became ubiquitous in many healthcare areas (administration, data management, analytics, and medical imaging). These systems have primarily supported clinical operations such as hospital data management (electronic health records, EHRs), medication dosing, rule-based alarms for continuous monitoring in ICUs [126], and ECG devices capable of automated interpretation [127].

The sluggish translation of automated decision tools into routine care highlights the intrinsic complexity of clinical decision-making and the difficulty of converting AI research into robust products [128]. Still, over the past decade machine learning (ML) has reignited both public and private interest in AI [129]. Beyond theoretical progress, breakthroughs in basic science (genome sequencing, gene editing) and rapid advances in software and hardware, including new programming languages and frameworks, have facilitated and accelerated development. Reference [130] explores the linkage between medical diagnosis and ML/AI in detail.

5.1.2. Logical and Mathematical Aspects of Medical Diagnosis in Complexity Science

From a computer science perspective, medical diagnosis exemplifies emergent behavior within complex adaptive systems (characterized by nonlinear interactions, uncertainty, and interdependent variables). Tools such as formal logic, probabilistic frameworks, and causal inference can enhance diagnostic accuracy (by integrating multiple patient data streams and

enabling the dynamic evolution of hypotheses). Precise diagnosis underpins effective healthcare, but misdiagnosis affects approximately 10–15% of cases worldwide [131–134], inflicting patient harm and costing up to \$100 billion annually in the U.S. [131–134].

Diagnostic reasoning

Diagnostic reasoning is an evolving process in which data are collected, analyzed, and hypotheses formulated. Clinicians perform active diagnosis (akin to “active listening”) to identify symptom causes — Pierce’s “Select and Test” model has been adapted for medical contexts [135–138]. Diagnostic reasoning blends cognitive operations, metacognitive skills, and formalized knowledge [139].

The dual-process theory differentiates fast, intuitive Type 1 reasoning (pattern recognition rooted in experience) from slower, deliberative Type 2 reasoning (employed when memory-based judgments fail or to correct errors); Type 2 also guides test selection [139]. Symbolic logic contributes to structured diagnostic reasoning [140]. Active diagnosis takes medical evidence E and yields disease hypotheses H , diagnostic tests T , and a stopping rule A :

$$f: E \rightarrow O^{|H|}, O^{|T|}, A, \quad (5.1)$$

where O is an ordered set (e.g., $R [0, 1]$) and A is a stopping criterion. Passive diagnosis reduces this to one-step classification:

$$f: X \rightarrow O^{|H|}. \quad (5.2)$$

Pattern-recognition algorithms applied to EHRs or medical images are passive — they can be highly accurate but often lack interpretability [141, 142]. Active systems like QMR/INTERNIST-1 incorporate sequential information gathering and defined stopping criteria [143]. Machine learning models, frequently passive, may learn “shortcuts” that impair generalizability, as illustrated in hip fracture and COVID-19 diagnostic studies [144–147]. Harmonizing active and passive approaches while correcting model errors and aligning with clinical guidelines remains challenging.

Diagnostic algorithms are intended as decision support to assist clinicians and reduce errors; yet, complexity in practice frequently limits integration (only basic systems are widely adopted [148]). Common diagnostic errors include search satisfaction (SoS), where clinicians stop searching after finding an abnormality — implicated in about 22% of radiology misreads [149]. Cognitive biases such as anchoring and premature closure account for roughly 10% of errors. Image segmentation and classification algorithms can detect anomalies and, in some cases, outperform human experts [150].

Knowledge-based diagnosis

The structured character of medical knowledge lends itself to knowledge-based approaches (logical statements within a knowledge base, KB). Although data-driven KBs are emerging, expert curation remains important [151]. Logic-based systems can be crisp (rule-based) or fuzzy (model-based).

Rules-based diagnosis

Rule-based systems implement “If A, then B” rules in decision-tree structures that are transparent and straightforward to debug. They encode sequential diagnostic questioning but face scalability limits because of the sheer volume of rules required (e.g., MYCIN required six years to build [123, 152]; AI/RHEUM and CADIAG-1 are other examples [153, 154]).

Fuzzy logic systems

Fuzzy logic allows truth degrees between 0 and 1, accommodating ambiguity in medical knowledge [63]. Systems such as SPHYNX, CADIAGS-2, and MILORD employed fuzzy logic from the 1980s onward [155–157]. Hybrid neuro-fuzzy systems combine fuzzy logic with neural networks to enhance performance [158].

Ontology-based systems

Ontologies represent medical knowledge as concepts and relations (e.g., “Respiratory virosis is an infectious disease”), enabling richer, model-based inference compared with sequential rule systems [159]. ICD11 and UMLS ontologies are integrated in EHR systems for tasks like link prediction [160, 161].

Model-based diagnosis

Model-based diagnosis (introduced in the late 1970s) decouples domain knowledge from diagnostic rules and models pathophysiological relationships between conditions and symptoms [162, 163]. Diseases are inferred from deviations from expected behavior. Consistency-based diagnosis treats faults as logical inconsistencies but suffers high inference costs (analogous to NP-complete problems) [164].

Abductive diagnosis

Abductive diagnosis seeks the most plausible causal explanations for observed symptoms by employing first-order logic and causal models (e.g., the CASNET project [165, 166]). Hypotheses can interact in complex ways (for example, immune suppression may reduce fever

while influenza increases it); parsimony via set-coverage heuristics balances simplicity and explanatory power [167, 168].

Bayesian diagnosis

Bayesian methods apply probability theory to weigh competing hypotheses given evidence and priors: Bayes' theorem computes posterior probabilities:

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)}, \quad (5.3)$$

where E are findings, H is hypothesis, $P(E|H)$ is conditional probability, and $P(H)$ is prior. Naive Bayes assumes conditional independence to simplify computation but neglects inter-hypothesis interactions [143]:

$$P(E_1, \dots, E_N|H) = \prod_{i=1}^N P(E_i|H). \quad (5.4)$$

Early applications such as QMR-DT and CASNET used bipartite Bayesian networks [143, 168].

Causal reasoning for diagnosis

Causal models enable inferences about interventions and counterfactuals (e.g., a symptom would not occur if the disease were absent). Counterfactual reasoning can enhance diagnostic accuracy by ignoring non-causal hypotheses, blending Bayesian, abductive, and consistency-based strengths, and frequently outperforming human experts in differential diagnosis [168, 169].

Machine learning for diagnosis

Machine learning (ML) methods identify salient patterns and derive decision rules from data. Unlike rule-based, hypothesis-driven methods, pattern-recognition approaches do not follow a sequential testing of causal hypotheses; instead, they search historical data for the closest match. ML has given rise to many algorithms (decision trees, SVMs, deep learning, etc.). Interest in data-driven techniques resurged in the 1990s, beyond traditional applied statistics [116]. ML approaches will be examined further in subsequent sections.

5.1.3. The diagnosis versus prognosis problem

The foundational goal of artificial intelligence (AI) described in [170] is to enable computers to discover patterns across heterogeneous, high-dimensional data that would otherwise go

unnoticed. Historically, realizing this aim required large computational resources and time, and was feasible only in areas like physics or astronomy. However, exponential growth in computing power and portability has broadened AI's applicability to domains such as critical care medicine, which face rich, complex datasets. Increasingly, ICU investigations apply AI-driven models. Yet, our understanding of AI's capabilities and practicality in critical care remains nascent. Moreover, several technical and methodological barriers must be surmounted before AI becomes an integral element of routine clinical practice.

Identifying the cause of clinical deterioration can be difficult because early disease progression is often subtle, and comorbid conditions can obscure the primary pathology (Figure 5.1). Correctly interpreting the clinical context is frequently demanding. For example, pulmonary infiltrates should not be assumed automatically to reflect alveolar fluid overload; potential causes include cardiogenic pulmonary edema, pleural effusion, parapneumonic fluid accumulation from infection or inflammation, or, in trauma, hematoma formation.

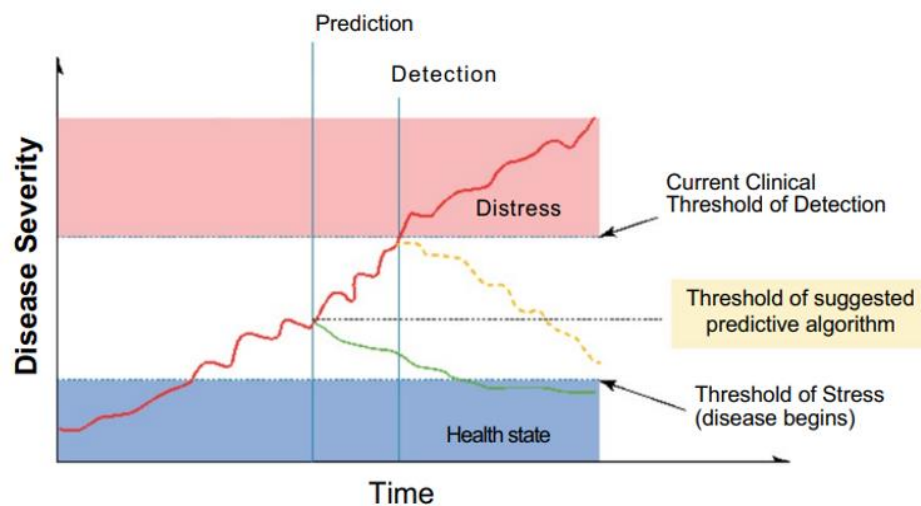


Figure 5.1. Conceptual meaning of AI-based predictive analytics in diagnosing and predicting the progression of diseases. Adapted from [170].

The AI model enables earlier detection or forecasting of disease so clinicians can intervene sooner with critically ill patients (green line) rather than later (yellow line).

Insufficient clinical context or lack of confirmatory testing may delay timely management. AI could mitigate such delays by improving diagnostic accuracy through text and image processing - for instance, a machine-learning model [171] can differentiate congestive heart failure (CHF) from other pulmonary pathologies. Detecting and forecasting disease trajectories is especially valuable for critically ill patients.

Because disease progression is continuous, a patient's clinical stability may follow diverse trajectories even before ICU admission [172]. In a study of step-down unit patients, those with

cardiorespiratory instability (hypotension, tachycardia, respiratory compromise, or hypoxemia) were more likely to deteriorate within 90 minutes; a dynamic random forest classifier produced personalized risk trajectories for each patient [173].

Rapid clinical decompensation is common in the ICU and, if unrecognized early, can result in mortality or irreversible sequelae. Hence, researchers have attempted to predict hemodynamic collapse. A normalized dynamic risk-score trajectory from a random forest model predicted tachycardia about 75 minutes before onset (a precursor to shock) [174]. Hypotension has also been forecasted intraoperatively [175].

Sepsis, a frequent ICU syndrome, has attracted considerable AI attention [141, 170, 176]. Generic, one-size-fits-all approaches falter for complex conditions. Despite extensive research over the past decade, outcomes in septic shock have not conclusively improved despite guideline implementation [177]. AI may partially address sepsis heterogeneity, variable etiologies, and diverse host responses by leveraging reinforcement learning (RL) to create individualized treatment strategies: RL models represent patient state via multiple variables, formulate action models, and learn from rewards and penalties linked to outcomes. Applied to sepsis, RL can suggest sequential treatment policies optimized for the patient. This exemplifies how AI can produce personalized clinical strategies with tangible effects [178]. For mechanically ventilated patients, time-series features (44 variables) were used with an RL (Markov Decision Process) approach that outperformed physicians' usual care in measures such as 90-day and ICU mortality [179].

These examples illustrate AI's potential to inform care for critically ill patients. AI-based decision aids could be especially impactful in settings with limited clinician availability or constrained resources (remote or underserved locations). Nevertheless, because AI recommendations are typically derived from retrospective observational data, any recommended policies must undergo rigorous analysis and prospective validation before clinical deployment.

For an AI system to be clinically useful in real settings, it must deliver substantial information rapidly. For critically ill patients, acceptable feedback times are extremely short (under a few minutes). Early predictions offer model-building time but often with limited predictive power; predictions closer to the event are more accurate but leave little time for data curation or model adjustment. Thus, practical real-time AI for critical care requires highly efficient data preprocessing pipelines and the ability to update models quickly with new inputs; outputs must be delivered at the bedside. Although current technological constraints limit fully-realized real-time AI implementations, such models could nonetheless yield substantial diagnostic and therapeutic benefits.

AI development has accelerated in step with unprecedented growth in computational capabilities. A wide array of critical-care research has harnessed AI to reveal hidden disease patterns in heterogeneous clinical datasets. AI models offer methods for disease detection, phenotyping, and prognosis, potentially changing the course of critical illnesses and enabling personalized treatment decisions when multiple therapeutic options exist [170].

5.1.4. Medical aspects of the epilepsy and sepsis research field

Epilepsy

Epilepsy is a neurological disorder characterized by recurrent, unpredictable seizures. It is among the most prevalent neurological conditions (WHO [180]), affecting more than 50 million people globally (second only to stroke). Seizures are the cardinal manifestation, occurring without warning, impairing quality of life, and increasing risks of injury and, occasionally, death.

About 30% of patients have drug-resistant epilepsy — seizures that do not respond to medications, and for whom surgery may also fail. For these patients, seizure-forecasting tools are desirable so they can avoid hazardous activities and reach safety prior to an event.

Electroencephalography (EEG) is a standard diagnostic method to detect and predict seizures, based on analysis of cerebral electrical activity. EEG-based seizure detection remains an active research domain with many proposed techniques. Several studies support the existence of a preictal state preceding seizures, characterized by consistent EEG patterns [181, 182].

EEG analysis techniques can be categorized variously. A review [183] groups methods into four major classes: (1) time-domain, (2) frequency-domain, (3) time–frequency-domain, and (4) nonlinear approaches. Modern work also employs machine learning. Within these categories sit mathematical tools such as linear prediction (LP), independent component analysis (ICA), fast Fourier transform (FFT), autoregressive (AR) methods, short-time Fourier transform (STFT), and wavelet transform.

Given that EEG signals are non-stationary and often noisy, purely linear analyses may be inadequate. Nonlinear dynamical methods have proven effective for characterizing these physiological signals. They can outperform traditional time- or frequency-domain approaches because they capture nonlinearity. Examples include higher-order spectral (HOS) techniques, phase space plots (PPS), correlation dimension (CD) and fractional dimension (FD) estimators, largest Lyapunov exponent (LLE) analyses, and entropy-based measures.

Entropy estimators (spectral entropy SEn, approximate entropy ApEn, sample entropy SampEn, etc.) fall under nonlinear methods and are largely grounded in Shannon entropy, also used to approximate algorithmic complexity (AC) [183, 184]. Emerging work questions whether

Shannon entropy is the optimal measure of AC; some advocate for the Kolmogorov–Chaitin definition of algorithmic complexity [184].

Classifiers applied to EEG data include k-nearest neighbors, support vector machines, random forests, multilayer perceptrons, convolutional neural networks, recurrent neural networks, and LSTM networks to distinguish preictal from interictal states. Nonetheless, reliably detecting preictal transitions with high sensitivity and specificity remains difficult [181].

In this study, algorithmic complexity (in the Kolmogorov–Chaitin sense) is employed as a feature and data representation technique during preprocessing before applying machine learning algorithms.

Sepsis

Sepsis arises when a severe infection provokes organ dysfunction and can be life-threatening. Infectious disease has been a central cause of mortality throughout human history; sepsis denotes a maladaptive host response to infection [185].

The term “sepsis” originates from the Greek “σηψις”, meaning decomposition of organic matter in the presence of bacteria [186]. Homer referenced sepsis in a medical context some 2,700 years ago, derived from the verb “σηπω”, “to putrefy.” While sepsis has been recognized since antiquity, its microbial etiology was not established until the 19th-century advent of germ theory.

In the early 1990s the first internationally agreed definition (“Sepsis-1”) described sepsis as systemic inflammatory response to infection [187]. Because systemic inflammation lacks specificity — occurring also in noninfectious conditions with better prognosis — “Sepsis-2” expanded the definition to include inflammatory response parameters, hemodynamic alterations, and organ dysfunction [188]. The 2016 Sepsis-3 redefinition characterized sepsis as a dysregulated, potentially fatal host response to infection, operationalized via the Sequential Organ Failure Assessment (SOFA): sepsis is suspected infection with an acute SOFA score increase of at least two points [185, 189].

Across history the core challenge has remained: early sepsis can be difficult to detect. Niccolo Machiavelli, writing in the 16th century, observed a similar clinical truth: a fever at onset is hard to recognize but becomes easier to identify — and harder to treat — as it progresses [190]. Modern therapeutics (targeted antimicrobials, immunomodulation) and monitoring tools exist, yet the fundamental issue persists: early, reversible organ injury is often clinically silent; each hour of delay in appropriate antimicrobials increases mortality [191]. Despite decades of study, sepsis continues to be a major public health burden with significant mortality, morbidity, and cost [192,

193]. Timely diagnosis and immediate initiation of appropriate antimicrobial therapy remain critical [194], yet clinicians struggle to identify sepsis in its earliest, reversible phase.

Symptoms are frequently nonspecific, and diagnostic uncertainty can delay urgent resuscitative and antimicrobial interventions that might be lifesaving [195]. The principal clinical screening tools employed to detect sepsis are SOFA, SIRS (Systemic Inflammatory Response Syndrome), and MEWS (Modified Early Warning Score). While imperfect, these scoring systems provide established performance criteria and serve as baselines for new predictive systems under development [191].

Machine learning-based systems (MLS) represent one such advancement [184]. Although MLS is a relatively recent adjunct in sepsis management, it holds promise for earlier detection and decision support in the ICU. MLS-based decision-support tools are rapidly evolving in sepsis research [196, 197].

5.2. Aspects of representation learning, complexity science, and machine learning related to research topic areas

5.2.1. Representation learning as a lens for current research

With the proliferation of high-throughput sequencing, data volumes have escalated rapidly [198]. These technologies generate vast datasets but do not inherently provide biological interpretation; thus, analyzing biological sequences (DNA, RNA, proteins) to extract meaningful biological information is increasingly important and demanding. One proposed solution has been to adapt natural language processing (NLP) techniques to sequence analysis, treating biological sequences as sentences and k -mers as words [199].

NLP seeks to enable computational systems to grasp human language content (including context), so they can precisely extract information and generate insights [200]. Natural language consists of characters (an alphabet) whose meaning arises from grammar and semantics. Biological sequences can similarly be viewed as strings of letters whose properties (function, structure) derive from biophysical and biochemical principles [201]. Like language, the meaning of a sequence element depends on its surrounding context — the word “bank” can refer to finance or riverbank depending on context; likewise, RNA secondary structure formation depends on neighboring sequence. Given these parallels, NLP methods have the potential to reveal function and structure embedded within biological sequences.

Representation learning [202] is a core NLP technique that enables systems to learn structural encodings of raw data (words or characters) [203]. Representations are typically real-valued vectors (distributed embeddings) that map semantically similar words to nearby vectors. Representation learning can directly produce vectorized encodings of biological sequences for

tasks such as function and structure prediction [204, 205]: high vector similarity between proteins often implies similar function and structure.

Vector similarity or distance is computed via linear-algebraic measures like dot product, Euclidean distance, and cosine similarity. Successful word encoding via representation learning is a major research focus [203], and downstream NLP and deep-learning performance depends substantially on representational quality. Accurate sequence representations are therefore crucial for clustering, function annotation, structure analysis, and disorder prediction.

Representation-learning approaches have been applied across domains relevant to this thesis (EEG/epilepsy [206–213], sepsis and related conditions [214–220]). Later chapters will discuss word-embedding techniques (Word2Vec, GloVe, fastText, BERT) and dimensionality-reduction methods and their relevance to the current research.

5.2.2. Contemporary research from the point of complexity science

An overview of complexity metrics (Zenil, 2023) [221] that are relevant to contemporary research can be summarized as follows:

- a) Statistical: Classical probability (P)
- b) Resource-wise: Computational (time) complexity
(asymptotic) ($O(n)$)
- c) Combinatorial: Shannon Entropy (H)
- d) Algorithmic: AP , K (asymptotic)
- e) Hybrid: Compression, Minimum Description Length (MDL)
Algorithmic + time (Bennett’s Logical Depth (LD))
- f) Fractal: Box-counting, Hausdorff dimension (D)
- g) Dynamic Mean-field measures, Lyapunov exponents, etc.

As noted in a recent survey [222], entropy-based measures are frequently used to quantify dynamic complexity in biological signals (EEG, ECG, electrodermal activity). Individuals may exhibit complex dynamics at different scales in these signals; transforming such dynamics into entropy-based features can aid in distinguishing groups. Entropy is widely used clinically, for example as a biomarker for seizure detection in epilepsy patients [223].

The most-used entropy estimators include approximate entropy (ApEn) [224], sample entropy (SampEn) [225], and fuzzy entropy (FuzzEn) [226]. ApEn embeds a short signal in phase

space and assesses how patterns change as embedding dimension increases, providing a measure of unpredictability. SampEn refines ApEn to reduce sensitivity to time-series length. FuzzEn replaces a strict similarity threshold (Heaviside function) with a fuzzy membership function (e.g., exponential family), better accommodating boundary uncertainty — advantageous for physiological data, particularly when signals are short or noisy [227, 228]. Some studies suggest FuzzEn may outperform nonfuzzy measures for physiological time series [223].

Differential entropy (an extension of Shannon entropy) has shown value as a feature for emotion recognition in EEG [229, 230]. Other approaches decompose signals into frequency bands or intrinsic mode functions (e.g., empirical mode decomposition) to extract more discriminative entropy features [223, 225, 228]. Alternative complexity measures (correlation dimension, Lyapunov exponents) have been proposed for ECG analysis [231].

Among these complexity measures, algorithmic metrics (AP, K) have received particular study using Algorithmic Information Dynamics (AID), a theoretical framework that integrates algorithmic information theory with dynamical systems analysis. The research described here extends this AID-based approach.

5.2.3. Machine learning techniques applied in medicine and medical research

Machine learning applications in medicine span many use cases and largely mirror general ML practice. Below is a concise taxonomy of frequently used methods (not exhaustive).

Machine Learning Paradigms:

- (a) supervised learning,
- (b) unsupervised learning
- (c) semi-supervised learning,
- (d) reinforcement learning, etc.,

applied to classification and regression depending on the question and data.

AI techniques:

- (a) classical ML;
- (b) Neural Networks/Deep Learning (NN/DNN – MLP, CNN, RNN);
- (c) Fuzzy-logic-based;
- (d) Bayesian-based, etc.

ML algorithms:

1. generalized linear models (GLM);
2. support vector machines (SVM);
3. k-nearest neighbors (k-NN);
4. tree-based methods (GB, DRF, etc.),
5. genetic algorithms, and more.

Medical domains with substantial AI/ML use:

- radiology [232],
- anesthesia and intensive care [197],
- pathology [233],
- cardiology [234],
- ophthalmology [235],
- logistics [236], among others.

In clinical practice in the USA, the FDA maintains a registry of AI/ML technologies that are in use. This confirms earlier observations: radiology remains the leading area of practical ML/AI adoption.

5.3. Medical prognosis using machine learning in epilepsy and sepsis

5.3.1. Representation learning and its application in research

This thesis principally investigates the application of representation learning in epilepsy and sepsis. The literature reports diverse representation-learning strategies in these domains (processing numerical and textual data), most of them experimental; reported performance (AUC) ranges approximately from 0.79 to 0.96.

5.3.2. Methods and algorithms used for classification purposes

Classification algorithms are central to ML-driven diagnosis and prognosis across medical domains (e.g., epilepsy, sepsis) because they are largely domain-agnostic. For methodological specifics, consult the cited studies employing statistical ML methods [237–239]. Diagnostic problems can often be framed as classification tasks, frequently reducible to binary decisions.

Supervised learning lies at the core of classification: models are trained on input–output pairs (e.g., patient-state vectors paired with diagnoses), each input associated with a label, and the aim is to learn a mapping function from inputs to outputs to enable prediction on unseen data. Let X denote the input vector space and Y the set of possible outputs or labels. There exists a joint probability distribution $P(X|Y)$ over the combined space, and n samples drawn from this distribution form the training dataset:

$$\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}, \quad (5.5)$$

where each x_i in the training set has a corresponding label y_i . The learning goal is to identify a function f from a hypothesis space H that minimizes a loss function $L(y, f(x))$ measuring discrepancy between predictions and true labels. The risk of hypothesis h is the expected loss:

$$R(h) = E[L(h(x), y)] = \int L(h(x), y) dP(x, y). \quad (5.6)$$

Minimizing this risk over

$$h^* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} R(h) \quad (5.7)$$

defines the learning objective. Choice of loss function depends on data type (continuous or categorical) and the task (regression or classification). Medical diagnostics frequently involve binary classification (disease present vs absent), and standard loss functions are applied (see synthesis papers [241–245]), which demonstrate that diverse ML methods stem from a core set of algorithmic families.

Notational conventions

- Inputs are X with components X_j , numerical outputs Y , and categorical outputs G (class labels). Observed values are denoted in lowercase (e.g., the i -th observation of X is x_i). Matrices are bold uppercase (e.g., X denotes an $n \times p$ matrix of input vectors); T denotes transpose.

- For binary classification, the target is encoded as $Y \in \{0,1\}$ and predictions in $[0,1]$ are mapped to classes by thresholding (e.g., 0.5). Multiclass outputs generalize this encoding.

Linear regression

Linear regression models a response variable as a linear combination of covariates (Figure 5.2) [237, 238]:

$$y = x^T \beta + \beta_0 + \epsilon, \quad (5.8)$$

where β is the parameter vector, β_0 the intercept, and ϵ a Gaussian Gaussian noise term. For an input vector $X^T = (X_1, X_1, \dots, X_p)$, the predicted output is:

$$\hat{Y} = \hat{\beta}_0 + \sum_{j=1}^p X_j \hat{\beta}_j. \quad (5.9)$$

The model is typically fitted by least squares, minimizing the residual sum of squares (RSS):

$$RSS(\beta) = \sum_{i=1}^N (y_i - x_i^T \beta)^2, \quad (5.10)$$

In matrix terms, with X the design matrix and y the response vector, the normal equations result by differentiating the loss with respect to β :

$$X^T(y - X\beta) = 0. \quad (5.11)$$

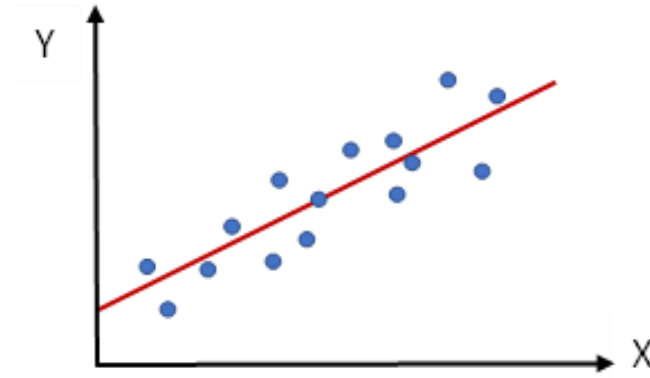


Figure 5.2. Fitting a line to data with linear regression.

In dichotomous classification, outputs are coded as 1 (disease present) or 0 (absent), and continuous predictions are thresholded to assign class labels. Adjusted continuous values are converted to class labels using a thresholding rule:

$$\hat{G} = \begin{cases} \text{disease PRESENT if } \hat{Y} > 0.5, \\ \text{disease ABSENT if } \hat{Y} \leq 0.5 \end{cases} \quad (5.12)$$

Regularized variants such as ridge regression and LASSO, which add penalty terms to improve generalization and robustness, are discussed elsewhere.

Tree-based methods

Tree-based approaches partition the feature space into axis-aligned rectangular regions and fit constants (e.g., means) within each partition (Figure 5.3).

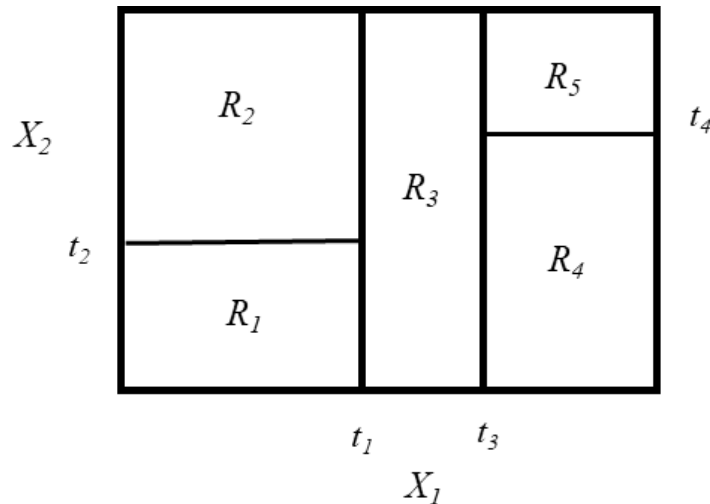


Figure 5.3. Partition of the feature space by lines parallel to coordinate axes. In each partition element Y can be modelled with a different constant (adapted from [237]).

Each partition element corresponds to a region R with an associated constant c equal to the mean of Y in that region; regression is applied region-wise:

$$\hat{f}(X) = \sum_{m=1}^M c_m I\{(X_1, X_2) \in R_m\}, \quad (5.13)$$

where variables X_1, X_2 in the feature set determine the risk I of misprediction or misclassification.

Based on the minimization of the sum of squares $\sum (y_i - f(x_i))^2$, we obtain the average value of y_i in region R_m :

$$\hat{c}_m = \text{ave}(y_i | x_i \in R_m). \quad (5.14)$$

Starting from an initial split, the tree is grown by recursive partitioning until stopping conditions (e.g., minimum node size) are met, yielding M terminal regions. Tree size is a tuning parameter optimized via cost-complexity pruning: grow a large tree T_0 and prune back to a subtree $T \subset T_0$ by removing internal nodes to balance complexity and performance. For classification, node labels are assigned by majority class; splitting and pruning decisions are guided by criteria such as:

$$\text{a) Misclassification error: } \frac{1}{N_m} \sum_{i \in R_m} I(y_i \neq k(m)) = 1 - \hat{p}_{mk}; \quad (5.15)$$

$$\text{b) Gini index: } \sum_{k \neq k'} \hat{p}_{mk} \hat{p}_{mk'} = \sum_{k=1}^K \hat{p}_{mk} (1 - \hat{p}_{mk}); \quad (5.16)$$

$$\text{c) Cross-Entropy or deviance: } - \sum_{k=1}^K \hat{p}_{mk} \log \hat{p}_{mk}, \quad (5.17)$$

where $\hat{p}_{mk}$ is the proportion of class k in node m . These criteria direct tree construction to maximize classification accuracy.

Artificial Neural Networks (ANN)

ANNs model complex, nonlinear functions and are organized in layers (Figure 5.4).

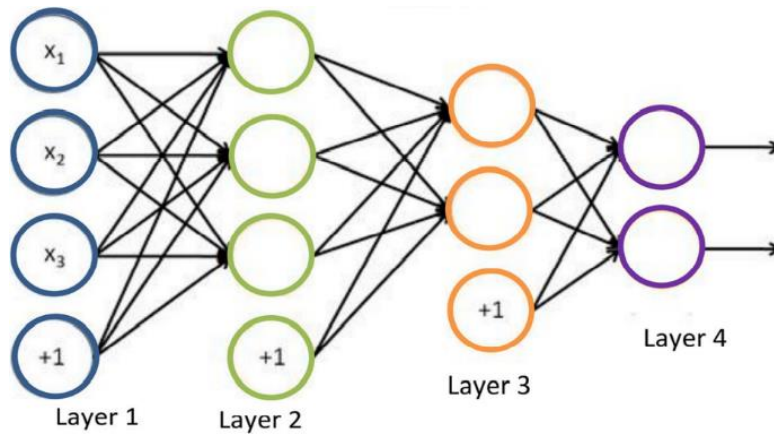


Figure 5.4. A fully connected (FC) artificial neural network (ANN) with four predictors and two hidden layers (adapted from [225]).

For input x , the first hidden layer computes activations:

$$z_m = \sigma(\alpha_{0m} + \alpha_m^T x), \quad (5.18)$$

where σ denotes a nonlinear activation (commonly ReLU: $\sigma(z) = \max(0, z)$).

These activations feed subsequent layers; the output layer then computes predictions:

$$\hat{y} = \beta_0 + \sum_{m=1}^M \beta_m z_m. \quad (5.19)$$

Parameters α and β are estimated by minimizing a squared error loss for continuous targets:

$$\sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (5.20)$$

Additional hidden layers extend model capacity; for example, a second hidden layer processes first-layer activations z_m :

$$z'_k = \sigma(\gamma_{0k} + \gamma_k^T z), \quad (5.21)$$

and the output layer uses a softmax function for classification:

$$P(y = k|x) = \frac{\exp(\beta_{0k} + \beta_k^T z')}{\sum_{j=1}^K \exp(\beta_{0j} + \beta_j^T z')}. \quad (5.22)$$

Parameters are learned via backpropagation.

Recurrent Neural Networks (RNNs)

RNNs handle sequential inputs $x_1, x_2, \dots, x_L$ and produce outputs y (Figure 5.5). Hidden states update recurrently:

$$h_l = \sigma(Wx_l + Uh_{l-1} + b), \quad (5.23)$$

with shared weights and biases W, U , and b , and outputs are computed as:

$$\hat{y}_l = Vh_l + c. \quad (5.24)$$

For binary outputs, a sigmoid activation may be applied. Parameters are optimized by minimizing sequence losses:

$$\sum_{i=1}^n \sum_{l=1}^L (y_{il} - \hat{y}_{il})^2. \quad (5.25)$$

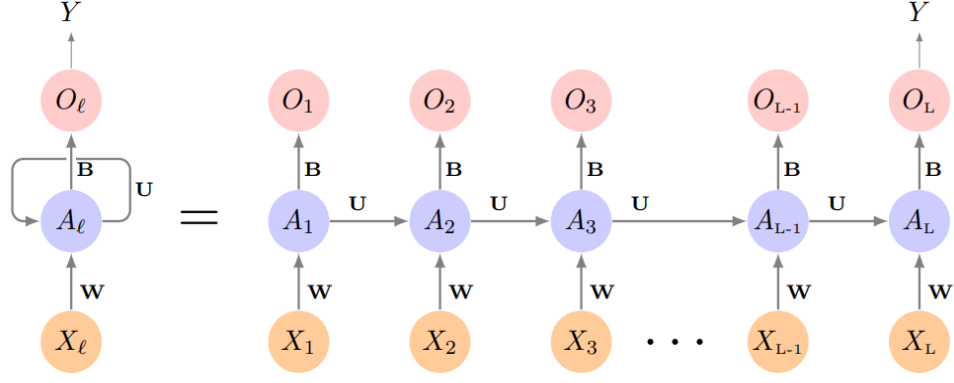


Figure 5.5. Schematic of a basic RNN (adapted from [237]).

RNNs retain memory of past inputs via hidden state updates, making them suitable for time series and language tasks.

Support Vector Machines (SVM)

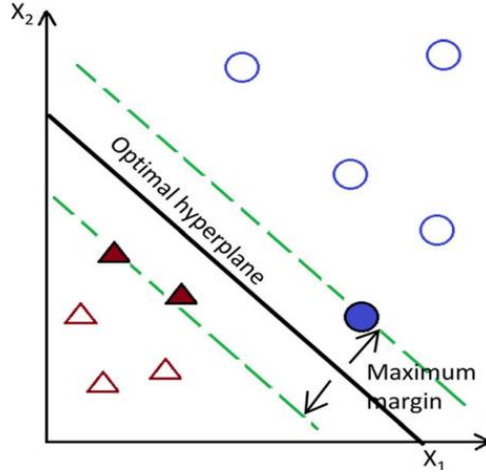


Figure 5.6. Support vector machine (SVM).

SVMs separate data in an n -dimensional space using a hyperplane:

$$\beta_0 + \beta^T x = 0, \quad (5.26)$$

Where correctly classified points x_i satisfy $y_i(\beta_0 + \beta^T x_i) > 0$ for correct classification (Figure 5.6) [237].

The classifier is:

$$f(x) = \sum_{i=1}^n \alpha_i y_i K(x, x_i) + b, \quad (5.27)$$

with K , a kernel function (e.g., linear: $K(x, x') = x^T x'$, polynomial, radial basis, etc.). Estimation relies on inner products between training observations, enabling robust classification in high-dimensional feature spaces [237].

Logistic Regression

Logistic regression maps inputs to binary probabilities using the logistic function and chooses coefficients to minimize classification error [211]:

$$P(y = 1|x) = \frac{1}{1 + \exp(-(\beta_0 + \beta^T x))}. \quad (5.28)$$

Multinomial logistic regression generalizes this to multiple classes. Logistic models are often more interpretable than deep neural networks.

Convolutional Neural Networks (CNNs)

CNNs are tailored for image tasks: convolutional layers apply learnable filters to detect local features (edges, textures):

$$(I * F)(i, j) = \sum_m \sum_n I(i + m, j + n) F(m, n), \quad (5.29)$$

followed by nonlinearities (ReLU), pooling (e.g., max pooling) to reduce spatial dimensions, and finally flattening and fully connected layers with softmax activation for class probabilities:

$$P(y = k|x) = \frac{\exp(z_k)}{\sum_{j=1}^K \exp(z_j)}. \quad (5.30)$$

Stacked convolutional/pooling layers progressively build higher-level representations, making CNNs powerful for complex image classification.

K-Nearest Neighbor (KNN)

KNN assigns class labels based on the majority label among the K nearest neighbors in feature space (typically using Euclidean distance):

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^p (x_{ir} - x_{jr})^2}. \quad (5.31)$$

KNN is simple but computationally intensive on large datasets due to distance computations.

Naïve Bayes (NVC)

Naïve Bayes computes class probabilities via Bayes' theorem under the simplifying assumption of feature independence:

$$P(y = k|x) = \frac{P(x|y = k) P(y = k)}{\sum_j P(x|y = j) P(y = j)}. \quad (5.32)$$

It is efficient and robust for categorical tasks with limited classes.

5.3.3. State-of-the-art examples and open research issues

Researchers employ a wide variety of methods for predicting and classifying epilepsy and sepsis. Often traditional algorithms (see sections 5.2.3, 5.3.1, 5.3.2) are used alone or in combination.

Below are recent high-performance examples from the last three years. Comparisons in sepsis are aided by the International Guidelines for the Management of Sepsis and Septic Shock; for epilepsy, high-performing results are summarized based on recent literature reviews.

A major challenge in both fields is heterogeneity in sample sizes, patient cohorts, methodologies, and reporting formats across studies, complicating direct comparisons.

Epilepsy

A recent review of deep-learning approaches for seizure detection reports an increasing number of high-performing EEG classification studies (e.g., over 25 papers in 2019 reporting accuracies between 84% and 100%) [246]. Table 5.2 summarizes such systems reported between 2020 and 2025.

Table 5.2. A summary of publications on high-performance EEG classification and prediction systems (2020–2025)

Nr.	Author(s)	Publication Year	Accuracy/ sensitivity* (%)	Source
1.	Cao J. et al.	2020	99.33	[246]
2.	Zhao X. et al.	2020	89.28	[248]
3.	Abiyev R. et al.	2020	98.67	[249]
4.	Geng M. et al.	2020	98.91	[250]
5.	Liang W. et al.	2020	99.0	[251]
6.	Lekshmy et al.	2021	98.0	[252]
7.	Usman S. et al.	2021	94.2*	[181]
8.	Chandu Y.S. and Fathimabi S.	2021	99.0	[253]
9.	Epmoghaddam D. et al.	2022	99.05	[254]
10.	Nazari J. et al.	2022	98.52*	[255]
11.	Yan J. et al.	2022	96.01*	[256]
12.	Mir W. et al.	2023	99.7	[257]
13.	Xu X. et al.	2023	0.91*	[258]
14.	Ibrahim A. et al.	2023	74-100*	[259]
15.	Qiao W. et al.	2024	0.94	[260]
16.	Lucasius C. et al.	2024	1.0	[261]
17.	Zhang F. et al.	2025	0.94	[262]
18.	Xiang J. et al.	2025	0.98	[263]

Table 5.3 lists several recent high-performance sepsis-prediction systems (2020–2025).

Table 5.3. High-performance systems for sepsis prediction (2020-2025)

Nr.	Author(s)	Publication Year	AUC ROC/accuracy*	Source
1.	Wickramaratne D., Mahmud S.	2020	0.97	[264]
2.	Kok C. et al.	2020	0.98	[265]
3.	Helguera-Repetto, A. et al.	2020	0.944	[266]
4.	Misra D. et al	2021	0.9483	[267]
5.	Sadasivun S. et al.	2022	0.93*	[268]
6.	Ke L. et al.	2023	0.93-0.99	[269]
7.	Laxmi L. et al.	2023	0.94	[270]
8.	Dalal S. et al.	2024	0.96-0.99	[271]
9.	Upadhyaya D. et al.	2024	0.83-0.90	[272]
10.	Siri D. et al.	2025	0.99	[273]

Note: AUC ROC – area under the ROC curve.

Direct comparison of these systems is difficult: top-performing sepsis models (entries 1 and 2) lack full methodological details (especially validation); system 3 focuses on a particular cohort (neonates); and sample sizes range widely. For instance, Ke L. et al. used gene-expression data from 916 sepsis patients and 125 healthy controls; Dalal S. et al. used 18 lab tests plus two demographic variables in non-ICU patients; Siri D. et al. predicted mortality in sepsis rather than sepsis onset.

In sepsis, these ambiguities can be partly addressed through reference to the “International Guidelines for the Management of Sepsis and Septic Shock” (November 2021) [274], which note that machine learning “can improve the performance of screening tools.” A meta-analysis of seven studies comprising 42,623 patients reported pooled predictive performance for nosocomial sepsis with AUC 0.89 (95% CI, 0.86–0.92), sensitivity 81% (95% CI, 80–81), and specificity 72% (95% CI, 72–72).

The ethical governance of medical AI must ensure that systems for diagnosis, triage, and treatment support are kept subordinate to accountable clinical judgment. Effective frameworks include prospective validation, continuous post-deployment monitoring, and clear liability for developers, institutions, and clinicians. Transparency around training data, performance limitations, and failure modes is important to mitigate bias and protect vulnerable populations. Regulation must be flexible enough to stay abreast of adaptive models without snuffing out useful innovation, especially where resources for independent oversight are limited.

5.4. Synthesis of Chapter 5

This chapter outlines the transition of medical diagnosis and prognosis from empirical practice to approaches informed by complexity science, representation learning, and machine learning. It reviews historical developments, mathematical underpinnings, the interplay between diagnosis and prognosis, domain-specific contexts (epilepsy and sepsis), and integration with modern computational techniques, highlighting implications for intensive care.

Beginning with a historical lens, Hippocrates (ca. 400 BC) helped formalize the notion of disease (Table 5.1). Technological and conceptual milestones include the light microscope (1660s), the stethoscope (1810s), X-rays (early 1900s), the electrocardiograph (1906), Turing's universal machine (1936) laying computational foundations, the Dartmouth conference and the birth of AI in the 1950s, rule-based expert systems in the 1980s, waning interest in the 1990s–2000s, and resurgence with deep learning and nascent quantum computing in the 2010s. Regulatory frameworks for software as medical devices emerged around 2018. While early medical practice relied on codes (e.g., Hippocratic Oath), formal symbolic logic and probability-based decision-making arose in the 1950s. Statistical methods dominated until the 1970s; AI-driven expert systems gained prominence in the 1980s as CDSS. The Internet catalyzed medical data management in the 1990s, and ML has recently reenergized AI research due to advances in handling complex data (e.g., genomic sequencing).

Mathematically and logically, diagnosis involves inferring disease from signs and symptoms, with misdiagnosis rates around 10–15% and substantial clinical and economic burdens. Diagnostic reasoning integrates data collection, hypothesis generation, and testing, combining intuitive (Type 1) and analytic (Type 2) cognition. Active diagnosis yields ranked hypotheses, suggested tests, and stopping rules; passive approaches reduce problems to classification tasks but often sacrifice interpretability and generalizability. Knowledge-based systems (rule-based, fuzzy logic, ontologies), model-based and abductive approaches, Bayesian and causal reasoning frameworks, and ML-based pattern discovery each offer strengths and weaknesses).

The diagnosis–prognosis dichotomy in AI centers on early detection and forecasting in complex ICU settings, where comorbidities and subtle early changes complicate identification. AI can assist with early detection (e.g., distinguishing CHF from other lung disease) and with predicting deterioration trajectories, but real-time deployment requires fast preprocessing and model updating. Reinforcement learning shows promise for individualized sepsis treatment strategies, sometimes outperforming clinician care in retrospective analyses. Nonetheless, AI-derived recommendations must undergo rigorous prospective validation.

Two medical exemplars are epilepsy and sepsis. Epilepsy affects over 50 million people; ~30% are drug-resistant, necessitating seizure forecasting. EEG analysis benefits from nonlinear dynamical and entropy-based methods; algorithmic complexity (Kolmogorov–Chaitin) is proposed here for preprocessing. Sepsis, redefined over time (Sepsis-1 to Sepsis-3), remains a major public-health challenge where early recognition and prompt therapy are crucial; ML systems increasingly enhance screening and prognosis.

Representation learning treats biological sequences as language, enabling embeddings (Word2Vec, GloVe, fastText, BERT) to capture functional and structural properties. Complexity metrics span statistical, combinatorial, algorithmic, fractal, and dynamic categories; entropy measures (ApEn, SampEn, FuzzEn) are widely applied in physiological signal analysis. Algorithmic Information Dynamics combines algorithmic information theory with dynamical systems and forms a basis for this research.

For prognosis in epilepsy and sepsis, representation learning shows promise (reported AUCs ~0.79–0.96). Classification frameworks frame diagnostic tasks as risk-minimization problems, utilizing linear models, tree-based methods, ANNs, RNNs, SVMs, CNNs, KNN, and Naïve Bayes as appropriate to data modality and task.

State-of-the-art reports suggest very high performance in some epilepsy deep-learning studies (accuracies near 99–100%), and sepsis prediction systems report AUCs up to 0.99. However, variability in datasets, cohorts, and methodologies complicates direct benchmarking. International guidelines acknowledge ML’s potential to improve screening, with pooled analyses indicating AUCs around 0.89 for nosocomial sepsis prediction

In summary, this chapter demonstrates how complexity science, representation learning, and machine learning have progressively reshaped diagnostic and prognostic approaches — moving from historical empiricism to computationally sophisticated methods capable of reducing diagnostic errors, enabling earlier interventions, and personalizing care in intensive settings. Challenges remain in model interpretability, heterogeneity of data, and real-time implementation. Promising directions include hybrid systems, use of algorithmic complexity for preprocessing, and prospective clinical validation to close the gap between experimental promise and clinical practice, ultimately improving outcomes for critically ill patients.

6. FROM RAW DATA TO ACTIONABLE INSIGHTS: IMPUTATION, COMPLEXITY METRICS, AND LEARNING REPRESENTATIONS FOR CRITICAL CARE APPLICATIONS

6.1. Data sets used in the present research

This study utilizes several sets of data. This part of the research focuses on three datasets: (a) the "EEG Epileptic Set" [51]; (b) the "EEG Non-Epileptic/Healthy Set" [275]; and (c) the "Sepsis Set" [50]. Additional datasets will be presented in coming chapters (e.g., Chapter 8).

6.1.1. An overview of the data utilized in this part of the study

a) EEG Epileptic Set

The data in set (a) is sourced from the National Institutes of Health, the Epilepsy Foundation, and the American Epilepsy Society for the international competition "American Epilepsy Society Seizure Prediction Challenge" [51]. The competition allows the use of the data for educational purposes. The objective was to determine the best model for distinguishing between preictal and interictal iEEG clips. The competition was conducted on the Kaggle platform (Kaggle, 2014). Intracranial electroencephalography (iEEG) was obtained from a group of five dogs diagnosed with spontaneous epilepsy, as well as from two human subjects who were undergoing iEEG monitoring. The iEEG signal from canine subjects was collected using 16 electrodes at a sampling rate of 400 Hz. The voltages were referenced to the average of the group. The recordings span several months to a year and encompass as many as one hundred seizures in certain dogs. The human patients were sampled at a rate of 5000 Hz, with the recorded voltages being referenced to an electrode positioned outside the skull. The monitoring period lasted up to one week. A total of 16 channels were recorded for the canine subjects, with the exception of dog-5, which had 15 channels. The number of electrodes for humans, patient-1, and patient-2 were 15 and 24 channels, respectively. The data is structured into segments that last for ten minutes each, containing either preictal or interictal activity. The training data is organized into sequences that span one hour each, while the timing of the test clips is not known. The preictal period commences one hour before the onset of a seizure, lasting for a duration of five minutes, from 1:05 to 0:05 prior to the occurrence of the seizure.

The dataset comprises 8002 subsets, each representing a ten-minute multichannel EEG recording clip. There are 4077 segments in the training set, 3766 of which are interictal and 311 of which are preictal. The remaining 3925 segments make up the test set. The human dataset is divided into two subsets, with 128 samples used for training and 345 samples used for testing. The present study utilized data from dogs 1 and 2, as well as data from one human participant, as

detailed in the forthcoming sections. The ultimate machine learning system was constructed utilizing the Dog-2 dataset, as it is one of the largest and most evenly distributed datasets available. Specifically, it consists of 500 interictal samples and 42 preictal samples.

b) EEG Non-epileptic/Healthy set

The dataset (b) [275] comprises EEG recordings of 36 individuals who were in satisfactory health. The recordings were taken both before and during the execution of mental arithmetic tasks. The arithmetic task involved performing a sequential subtraction operation on two numbers. The trial commenced by orally presenting a 4-digit number (minuend) and a 2-digit number (subtrahend) for communication, such as 3141 and 52. The set will be referred to as the arithmetic-test set from now on.

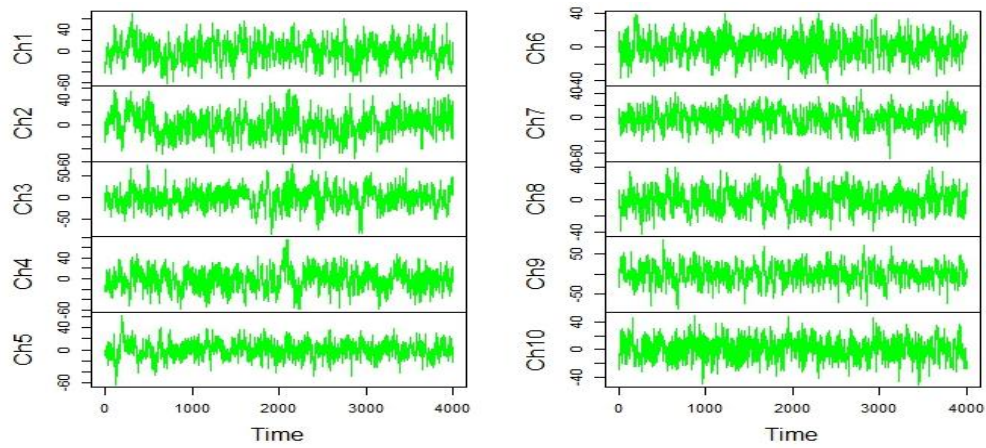


Figure 6.1. Visual representation of EEG data from ten electrodes (channels); representative for dataset (a) and dataset (b).

c) Sepsis set

The sepsis data utilized in this study consist of time series data collected at hourly intervals, encompassing 40 parameters, such as vital signs, laboratory indices, and logistics. The data sets consist of publicly accessible data [50]. These data originate from two separate US hospitals, which are geographically distinct from each other. The data was gathered over the past ten years with the consent of the institutional review boards, anonymized, and categorized based on Sepsis-3 clinical criteria. The dataset comprises more than 1.5 million time-windows and over 10.4 million data points (non-missing physiological and laboratory variables). Figure 6.2 illustrates the temporal changes in various parameters, including heart rate (HR), oxygen saturation (SaO_2), temperature (Temp), mean arterial pressure (MAP), respiratory rate (RR), base excess (BE), bicarbonate (HCO_3), fraction of inspired oxygen (FiO_2), lactate, and white blood count (WBC). There are a total of 40,336 patients or clinical cases in intensive care units.

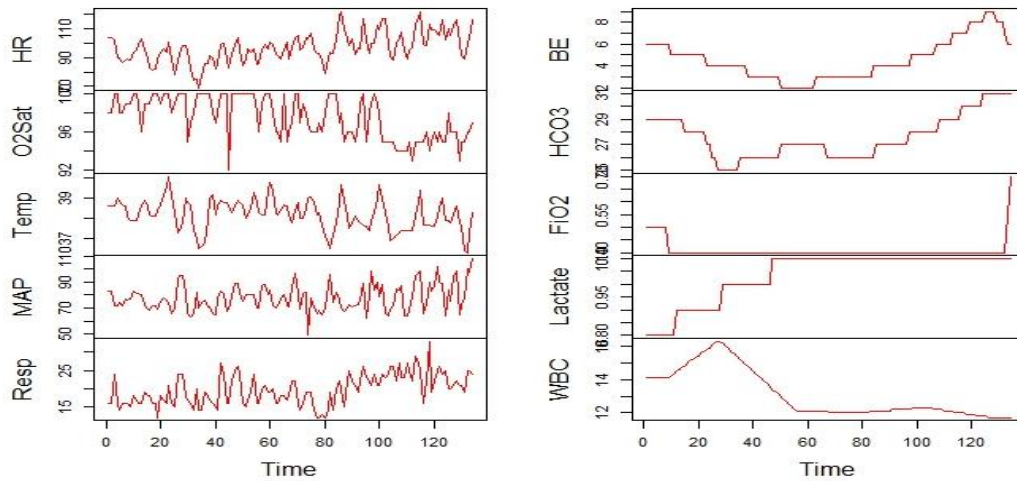


Figure 6.2. Visual representation of the raw sepsis data.

Table 6.1. Overview of the datasets employed in the study

Data characteristics	Set (a)—intracranial EEG signals from epileptic subjects	Set (b)—EEG and ECG signals from healthy subjects	Set (c)—multimodal data from patients in ICU
Number of cases in the original set	4 canines and 2 humans with epileptic seizures (total – 8002 subsets)	36 healthy volunteers passing an arithmetic test	40366 – patients (2932 with sepsis and 37434 with other pathologies)
Data format of a case	Multivariate time series as EEG signals from intracranial electrodes with sampling frequency of 400–5000 Hz	Multivariate multimodal time series as EEG and EKG, with sampling frequency of 500 Hz	40 parameters as multimodal time series condensed in hourly windows
Dimensions and format of the original data set	113.62 GB “.mat”	175 MB “.edf”	242 MB, “.psv”
Dimensions of a subset/case	7.67 MB (.mat) 11.2 MB (.csv)	1285–3883 KB (.edf)	2-55 KB (.psv) 5-38 KB (.csv)
Parameters/variables	EEG signals from 15–24 electrodes	EEG signals from 20 electrodes and an EKG derivative	40 parameters (vital signs, laboratory, demographics)
Particularities of the set	10-minute EEG clips/segments, no missing value	60-180 second EEG clip, no missing value	Rate of the missing values up to 80.9 %
The task solved within the thesis	Prediction/classification into (a) normal EEG segment vs (b) preictal	Representation learning	Prediction of (a) high risk of sepsis vs (b) low sepsis risk

Note: ICU refers to intensive care unit, ".edf" stands for European data format, ".psv" represents pipe-separated value, and ".csv" denotes comma-separated value.

The sepsis set A was chosen for further processing because it has fewer missing values. This dataset consists of 20,336 patients/cases, with 1,790 of them being sepsis cases. The average length of time that patients stay in the intensive care unit is 38.86 ± 22.31 hours. The data set we are considering has missing values, and some of the ways to handle this problem will be discussed in the next section. Table 6.1 summarizes a short description of the data sets used in the present study..

6.1.2. Preliminary data processing. Data gap filling

Investigations often face absent data, but few examine their possible impact on outcomes. Recent comprehensive evaluations [276, 277] have disclosed that many healthcare datasets are deficient and exhibit absent values. Consequently, it is essential to cleanse and substitute absent values to enhance the efficacy and precision of data analysis.

This section offers an exhaustive evaluation of the present condition of the discipline with an in-depth exploration of diverse methodologies for employing machine learning to address absent data. Considering that the EEG data employed in this research, as previously noted, is devoid of any missing values, the ensuing analysis chiefly focuses on the septic data set [50]. The distinction between random and nonrandom "missingness" in this situation is crucial to acknowledge. Nonetheless, when employing observable data, it can be difficult to differentiate between missing at random and missing not at random. Given that the dataset employed in this research [278] lacks an explanation for the absence of data, it is presumed that the missing information arises randomly (for instance, during a medical procedure requiring the removal of data collection sensors, human mistakes, equipment failures, etc.). The possible bias resulting from the recovery technique is thought to be diminished when data is absent at random, however data retrieval may still be feasible irrespective of the missingness type [279].

When faced with absent values in data, there are several methods to rectify them. Continuous data, like time series, requires fewer tools than static data, since most methods are relevant to the latter. The most straightforward approach for case analysis is complete case analysis (CCA) [280], which involves excluding observations that have missing values. Nonetheless, in numerous circumstances, especially when data is scarce, this method may prove impractical. This problem is even more important for imbalanced datasets, since each minority class occurrence is quite valuable for machine learning methods. Data recovery methodologies are often classified in the following manner:

- a) Single imputation is a technique for substituting a missing data point with one specific value. This methodology frequently incorporates methods such as Last Observation Carried Forward (LOCF), Baseline Observation Carried Forward (BOCF), and Next Observation Carried Backward (NOCB), and additional ones based on mean value or regression.
- b) Multiple recovery techniques employ repeated imputations to generate credible imputed data sets. This entails employing numerous credible imputed data sets and appropriately aggregating the outcomes from each. Numerous statistical software applications, including Amelia, FURIA, and MICE within the R programming language, are accessible for this objective.

According to specialized guidelines [281], excluding instances with missing data reduces the statistical power. The likelihood of a power reduction escalates with the rise in the quantity of absent data points. It is essential to exert effort to minimize the quantity of absent data. Unfortunately, a universally applicable systematic method for handling missing data does not exist. Consequently, in trial preparation, it is essential to devise ways to reduce missing data and identify the suitable method for its management throughout analysis.

This research delineates three techniques for data recovery, each comprising distinct procedures and components: (a) conventional approaches such as LOCF and NOCB; (b) a less prevalent technique utilizing Kalman filtering; and (c) regression imputation, which substitutes absent variables with forecasts derived from a regression analysis of available variables. The notion of viewing the human body as a complex system, wherein diverse characteristics influencing its operation (whether in health or disease) are interrelated, and where partial insights can be gleaned from existing data via rough correlation, constitutes a core principle of the ultimate method.

This segment of the research concentrates on delineating diverse techniques for the imputation of absent values, particularly highlighting their application in fulfilling prospective goals through the utilization of the restored data.

This research project seeks to develop a software program that employs machine learning to forecast sepsis, utilizing data gathered through many methodologies to construct this machine learning framework, which will be further upon in later parts.

The publicly available sepsis data came from two separate US hospitals. These data were collected during the past 10 years with Institutional Review Boards permission, anonymized and classed according to Sepsis-3 clinical standards. The dataset includes eight critical vital sign markers, 26 laboratory assessments, and six demographic variables. The compilation includes

hourly recaps of essential indicators, laboratory findings, and fixed patient profiles for an aggregate of 40,336 individuals. The attributes of each patient were summarized into hourly segments. For instance, several heart rate assessments conducted within an hour were consolidated as the median.

Set B comprises more than 80% of the absent items within its dataset. Set A was therefore chosen for further analysis due to lower percentage of missing data and higher prevalence of sepsis (8.80% vs 5.71% in set B). Among the total of 20,336 patients in this cohort, 1,790 individuals are diagnosed with sepsis. Within this cohort of septic patients, 502 subcategories have absent values for at least one of the six pertinent metrics. Following the implementation of the preliminary selection criteria, which necessitate a minimum of 7 hourly observations prior to the sepsis diagnosis and the exclusion of artifacts, there are 211 subgroups with incomplete data that may be recoverable.

Table 6.2 depicts a sepsis dataset featuring a single parameter (DBP) with entirely absent values (NA), whilst other physiological parameters exhibit partial missing values. It delineates the analysis of seven particular criteria, including age and labeling, which were chosen for more scrutiny. The parameters include 6 values representing vital signs and the patient's age.

Table 6.2. An excerpt from an original sepsis subset [278]

HR	O ₂ Sat	Temp	SBP	DBP	Resp	Age	Sepsis label
NA	NA	NA	NA	NA	NA	43.55	0
83	99	NA	109	NA	13	43.55	0
89	99	NA	102	NA	22	43.55	0
82	98	36.56	108	NA	20	43.55	0
91	98	NA	108	NA	16	43.55	0
93	99	37.1	106	NA	20	43.55	1
95	100	NA	130	NA	15	43.55	1
96	98	37.2	121	NA	18	43.55	1

Note: HR stands for heart rate, O₂Sat refers to peripheral blood oxygen saturation, Temp indicates temperature, SBP represents systolic blood pressure, DBP stands for diastolic blood pressure, and Resp denotes respiratory rate. The sepsis label refers to the classification of the observation as either 0 (non-sepsis) or 1 (sepsis). NA indicates a missing value. Data are reproduced from the author's article [278].

The data recovery performed in this study is based on a range of simple and commonly utilized algorithms/methods, both separately and in conjunction, as well as more specialized procedures tailored to the particular data involved. The first approach is often utilized for similar data, whereas the latter two techniques are expressly designed for the purpose of data recovery.

a) LOCF and NOCB

The Last Observation Carried Forward (LOCF) method is a strategy for addressing absent data by utilizing the most recent recorded value in a column to substitute the subsequent missing value(s). The Next Observation Carried Backward (NOCB) approach addresses absent data by utilizing the inverse of the Last Observation Carried Forward (LOCF) method. Examples of where these techniques may be applied encompass the columns designated as "HR," "O2Sat," "Temp," "SBP," and "Resp" in Table 6.2. These methods cannot be utilized for columns that include entirely absent values, exemplified by the "DBP" column in Table 6.2.

b) A personalized methodology

This methodology was created to address both situations: columns with incomplete missing data and columns with entirely absent values. This approach encompasses the subsequent steps when every column contains at least one value: (a) evaluating each column for absent data (NA); (b) "retrieving" the initial and terminal values (per column) by referencing the nearest cell in the same column; (c) finalizing interpolation by determining the values situated between two existing values, factoring in the trend (ascendancy or descendancy). The "na.approx()" function within the "zoo" package in R [283] serves the purpose of interpolation. An illustration of its application is evident in the "Temp" column of Table 6.2.

In instances where every value in a column is absent, the restoration process is as outlined: (a) The present values for each parameter/column are extracted independently by class (e.g., septic patients from set A) from the total cases set. The count of "non-missing" values (n) is ascertained, followed by the computation of the mean and standard deviation for these values. Gaussian distributions are produced for each of the six parameters utilizing the `{rnorm()}` function in R, with "n," "mean," and "sd" as inputs. The absent values in the initial dataset are subsequently substituted with values derived from these produced distributions.

c) A technique utilizing Kalman filtering alongside machine learning algorithms

This technique utilizes Kalman filtering for columns that have at least three non-null values. In instances of entirely absent values inside a column, generalized linear models are employed for data restoration. The Kalman filter, as presented in [284], employs a discrete filtering framework based on linear Gaussian dynamic and measurement models. This text provides a succinct overview of the core principles of this methodology, including aspects of optimal control and dynamic linear models as outlined in [285].

The Kalman filtering method is utilized to assess states in linear dynamic systems depicted in a state-space representation. It offers approximations of unidentified variables derived from temporal observations. This model posits that the state transitions from time $t - 1$ to time t in the following manner:

$$x_t = Fx_{t-1} + Bu_{t-1} + w_{t-1}. \quad (6.1)$$

The state transition matrix, referred as F , is derived from the preceding state vector x_{t-1} . The control-input matrix, referred known as B , is utilized with the control vector u_{t-1} . The process noise vector, represented as w_{t-1} , is presumed to follow a zero-mean Gaussian distribution with covariance Q , namely Q , i.e., $w_{t-1} \sim N(0, Q)$.

The connection between the present condition and the measurement is established by integrating the process model with the measurement model as outlined:

$$z_t = Hx_t + v_t. \quad (6.2)$$

The vector of measurements is represented as z_t , the matrix of measurements is represented as H , and the vector of measurement noise is represented as v_t . It is presumed that the measurement noise vector adheres to a zero-mean Gaussian distribution characterized by covariance C , denoted as $v_t \sim N(0, C)$. The Kalman filter is designed to provide an estimate of the state at time t , using the previous estimate x_0 , and the series of measurements and the system parameters specified by F, B, H, Q , and C . The user can modify the parameters Q and C to attain the preferred performance. The Kalman filter method consists of two separate phases: forecasting and revision. It can be concisely articulated as follows:

Expectation (phase one):

$$\text{Assessing the expected state} \quad \hat{x}_t^- = F\hat{x}_{t-1}^+ + Bu_{t-1} \quad (6.3)$$

$$\text{Calculating the projected error covariance} \quad P_t^- = FP_{t-1}^+ + F^T + Q \quad (6.4)$$

Revision (phase two):

$$\text{Determining the remainder of a measurement} \quad \tilde{y}_t = z_t - H\hat{x}_t^- \quad (6.5)$$

$$\text{Calculating the Kalman gain} \quad K_t = P_t^- H^T (R + HP_t^- H^T)^{-1} \quad (6.6)$$

$$\text{Amending the assessment of the condition} \quad \hat{x}_t^+ = \hat{x}_t^- + K_t \tilde{y}_t \quad (6.7)$$

$$\text{Adjusting the error covariance} \quad P_t^+ = (I - K_t H)P_t^-, \quad (6.8)$$

The "hat" operator means an estimate of a variable. Notes: Superscripts "-" and "+" denote expected (prior) and modified (posterior) estimations, respectively.

The data recovery method adopted in this work is based on the “imputeTS” package in R [286]. It is an advanced version of the above method (Equations 6.3–6.8) incorporating Kalman smoothing within structural time series models.

To ascertain the optimal technique for addressing absent values in columns with a significant amount of missing data, various methods were evaluated, including linear regression, exponential regression, and polynomial regression. Further details concerning this facet of the research are

available in Appendix 2. Ultimately, the Generalized Linear Model (GLM) methodology was chosen for more exploration due to the reasoning presented in the previously mentioned appendix.

The method utilized for retrieving data in columns with absent values is a Generalized Linear Model (GLM) [287] available on the H2O platform [288]. The GLM, integrated inside the H2O platform, employs the Gaussian method to depict the association between the response variable and the covariate vector x as a linear function:

$$y = x^T \beta + \beta_0 + \epsilon, \quad (6.9)$$

The parameter vector is symbolized by β , the intercept is represented as β_0 , and the model's noise ϵ is characterized by a Gaussian random variable $\epsilon \sim N(0, \sigma^2)$.

A model's estimate is found by maximizing the log-likelihood relative to the parameter vector, given the observed data. The Generalized Linear Model (GLM) [289] used in this study fits the model by optimizing the likelihood with parameter regularization:

$$\max_{\beta, \beta_0} (GLM \text{ Log} \cdot \text{likelihood} - \text{Regularization Penalty}). \quad (6.10)$$

The regularization penalty is determined by computing the weighted aggregate of the ℓ_1 (least absolute shrinkage) and ℓ_2 (ridge regression) norms of the coefficient vector. The equation for the regularization penalty is stated as follows:

$$\lambda P_\alpha(\beta) = \lambda \left(\alpha \|\beta\|_1 + \frac{1}{2} (1 - \alpha) \|\beta\|_2^2 \right). \quad (6.11)$$

The elastic net parameter is represented as $\alpha \in [0, 1]$, whereas λ serves as a tuning parameter that dictates the degree of regularization implemented in the elastic net model. It is crucial to acknowledge that the intercept term is not subjected to penalties in this approach. The optimization assignment related to observations can be articulated as follows:

$$\max_{\beta, \beta_0} \sum_{i=1}^N \log f(y_i; \beta, \beta_0) - \lambda \left(\alpha \|\beta\|_1 + \frac{1}{2} (1 - \alpha) \|\beta\|_2^2 \right). \quad (6.12)$$

The Gradient Boosting Machine (GBM), accessible on the H2O platform [288], proved to be the most proficient algorithm in the concluding phase of machine learning.

This study employs machine learning methods at different phases for two particular aims: (a) to impute absent values utilizing generalized linear models (GLMs) and (b) to develop the ultimate sepsis prediction model. A variety of algorithms were tested during this phase, with specifics outlined in the following sections. The primary processing phases are illustrated by a flowchart akin to the Consolidated Standards of Reporting Trials (CONSORT) in Figure 6.3.

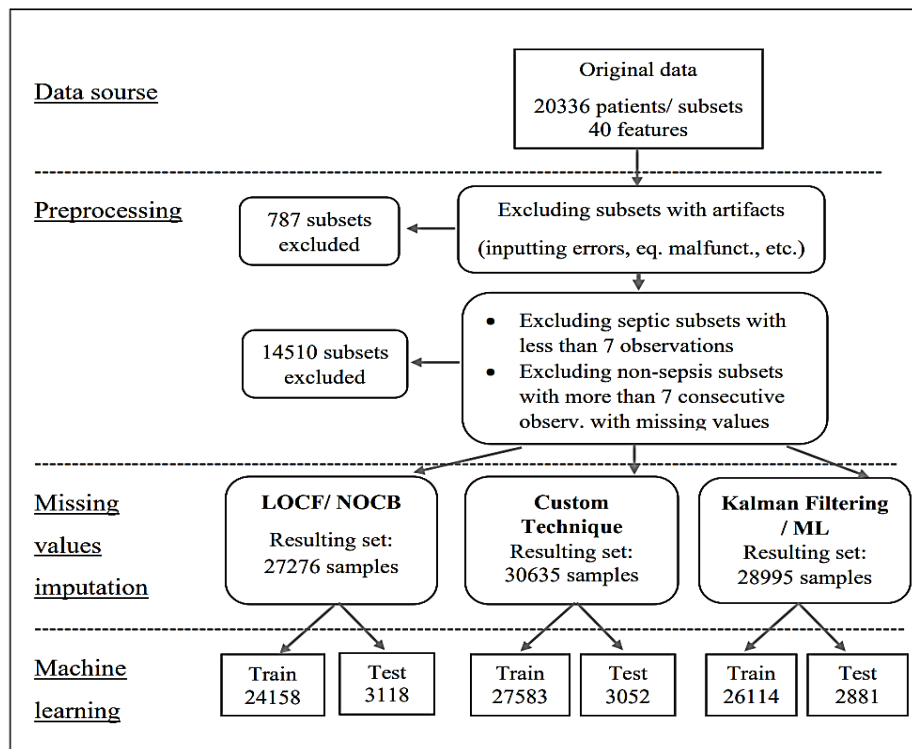


Figure 6.3. Data processing sequence in the present study (a CONSORT-style diagram).
Adapted from [278].

The programming language employed in this segment of the study is R. Multiple packages from the identical environment are employed for diverse purposes during the research. The visualization and engagement with the H2O ML platform are conducted utilizing the identical language and environment.

The data processing cycle of this research includes multiple techniques, such as missing values imputation, to produce datasets appropriate for machine learning.

At this phase, documents that encompassed artifacts like human mistakes, equipment failures, or malfunctions were excluded. Also, sepsis records with less than seven observations were excluded since the goal of the study was to build a sepsis prediction system with a prediction horizon of at least four hours, which required three-hour observations for the initial prediction. For the non-sepsis groups, a similar process was used, only keeping files with 7 or more consecutive observations with no missing data for the relevant parameters. This approach is shown in Figure 6.3.

At baseline, 502 sepsis subgroups have missing results for at least one key parameter. In the next phase, certain files were reconstructed by applying the custom technique mentioned before or by applying machine learning algorithms for data recovery as a part of the third approach that applies Kalman filtering.

The primary technique utilized in this study for data imputation is the LOCF (Last Observation Carried Forward) method, executed via the "ImputeTS" package in R [286]. The identical package is employed during the following phase, referred to as NOCB (Next Observation Carried Backward). This technique is appropriate for columns that have certain absent values. Nevertheless, it proves ineffectual for columns that are completely devoid of values, yet it may adeptly recover lost data in columns with partially missing values. The alternative (custom) method can accommodate both partially and entirely absent values in a column. Consequently, it generates a more extensive final collection of samples (30635 in total) that may be utilized to develop the prediction system. The third method, integrating Kalman filtering and machine learning, tackles the problem of columns containing partially and completely absent values. This method uses Kalman filtering to retrieve data in columns containing three or more non-null values.

To deal with the missing data problem, we investigated the association between the six specified factors and age. The variable age exhibited complete data availability and demonstrated a moderate association with several essential parameters. The three parameters with the greatest correlation coefficients were chosen for each of the six parameters. For instance, regarding temperature, the characteristics most closely associated are heart rate, systolic blood pressure, and age. The correlation coefficients for seven variables are presented in Table 6.3. The most important correlation coefficients are emphasized in bold typeface.

Table 6.3. Relations between data as coefficients of correlation [278]

	HR	O ₂ Sat	Temp	SBP	DBP	Resp	Age
HR	1.00	-0.12	0.18	-0.02	0.24	0.18	-0.18
O ₂ Sat	-0.12	1.00	0.02	0.06	0.02	-0.19	-0.02
Temp	0.18	0.02	1.00	0.09	0.03	0.08	-0.16
SBP	-0.02	0.06	0.09	1.00	0.53	0.07	-0.01
DBP	0.24	0.02	0.03	0.53	1.00	0.06	-0.33
Resp	0.18	-0.19	0.08	0.07	0.06	1.00	0.07
Age	-0.18	-0.02	-0.16	-0.01	-0.33	0.07	1.00

Note: HR (heart rate), O₂Sat (peripheral blood oxygen saturation), Temp (temperature), SBP (systolic blood pressure), DBP (diastolic blood pressure), Resp (respiratory rate), and Age. Data are reproduced from the author's work [278].

Several generalized linear models (GLMs) were formulated using the above correlation matrix. Models that yielded the best predictive ability were selected for deeper evaluation. The principal characteristics of those selected models are summarized in Table 6.4.

Table 6.4. Coefficients of linear regression for the parameters of significance [278]

Parameter for recovery	Intercept	Coefficient (Parameter)		
HR	89.4392	3.2658 (DBP)	3.2980 (Resp)	-2.3973 (Age)
O ₂ Sat	97.4894	-0.2305 (HR)	0.1840 (SBP)	-0.4954 (Resp)
Temp	37.2479	0.1233 (HR)	0.0720 (SBP)	-0.1001 (Age)
SBP	123.6418	1.6593 (Temp)	11.7255 (DBP)	0.6840 (Resp)
DBP	61.4999	2.5176 (HR)	6.8117 (SBP)	-3.7355 (Age)
Resp	20.3789	0.9001 (HR)	-1.0138 (O ₂ Sat)	0.3220 (Temp)

Note: HR stands for heart rate, O₂Sat refers to peripheral blood oxygen saturation, Temp indicates temperature, SBP represents systolic blood pressure, DBP stands for diastolic blood pressure, and Resp refers to respiratory rate. Data are reproduced from the author's work [278].

The missing-value imputation workflow combines Kalman filtering with the regression models above to reconstruct incomplete sepsis records. Table 6.5 shows an example of a recovered file produced by this pipeline, where the recovered entries are emphasized. The resulting filled-in subset corresponds to the dataset shown in Table 6.2.

Table 6.5. An illustration of recovered sepsis information [278]

HR	O ₂ Sat	Temp	SBP	DBP	Resp	Age	Sepsis label
83.37	98.75	36.56	108.33	60.22	17.12	43.55	0
83	99	36.56	109	60.37	13	43.55	0
89	99	36.56	102	59.09	22	43.55	0
82	98	36.56	108	59.93	20	43.55	0
91	98	36.83	108	61.2	16	43.55	0
93	99	37.1	106	60.87	20	43.55	1
95	100	37.15	130	68.45	15	43.55	1
96	98	37.2	121	65.86	18	43.55	1

Note. HR refers to heart rate, O₂Sat refers to peripheral blood oxygen saturation, Temp refers to temperature, SBP refers to systolic blood pressure, DBP refers to diastolic blood pressure, and Resp refers to respiratory rate. The sepsis label indicates whether the observation is classified as non-sepsis (0) or sepsis (1). Data are reproduced from the author's work [278].

From the sepsis records prepared as described above, samples for machine learning are generated. The following sections describe that procedure in detail.

6.2. Concepts and instruments of algorithmic complexity during the preprocessing phase

6.2.1. The Kolmogorov-Chaitin complexity and the Block Decomposition Method as fundamental methodologies for determining algorithmic complexity.

In this research, the assessment of algorithmic (Kolmogorov-Chaitin) complexity is crucial in data processing workflows [25] and is executed by the Block Decomposition Method. The algorithmic complexity, also called Kolmogorov-Chaitin complexity or program-size complexity,

is defined as the length of the shortest program, p , which outputs the string s on a universal Turing machine T [35]:

$$K_T(s) = \min\{|p|, T(p) = s\}. \quad (6.13)$$

Algorithmic Information Dynamics (AID) is an evolving subfield of complexity science grounded in algorithmic information theory. It encompasses concepts such as Kolmogorov-Chaitin complexity, algorithmic probability, data compression, ideal inference, the universal distribution, and Levin's semi-measure, among others [109]. AID aims to investigate the underlying causal mechanisms—why certain initial conditions lead to particular outcomes—distinguishing it from conventional statistical approaches. Practically, AID uses computational methods to build mechanistic models that illuminate causation and the fundamental rules governing physical processes, and it supports advances in machine learning.

The AID toolkit includes a specialized utility known as the online algorithmic complexity calculator (OACC) [290]. OACC is designed to provide accurate approximations for otherwise intractable functions and is highly effective at estimating algorithmic complexity and algorithmic probability for short and long strings as well as two-dimensional arrays. OACC captures algorithmic information beyond mere statistical regularities by applying the BDM technique, which relies on algorithmic probability and the coding theorem method (CTM) [109, 290]:

$$BDM = \sum_{i=1}^n CTM(block_i) + \log_2(|block_i|). \quad (6.14)$$

The OACC is accessible online and as standalone libraries in several programming languages, including R, Python, and the Wolfram Language, and it was used for specific calculations in this project.

At this stage of processing, every file in the dataset is a 'csv' subset; these files represent either EEG voltage traces across n channels over 10 minutes (EEG data) or hourly recordings of 40 sepsis-related variables. The EEG subset is augmented into a matrix where columns correspond to the n channels (electrodes) and rows to temporal observations of signal variation per channel.

6.2.2. Assessment of the Kolmogorov-Chaitin complexity concerning epilepsy dataset information

The EEG data matrix is partitioned into many small matrices of size 16×16 (or 15×15 , depending on the channel count), preserving the spatial-temporal relationship between electrodes and time. These small matrices are binarized using the BASCA approach and the "Binarize" function in R. BDM values are computed for each small block and then ordered along the temporal

axis to produce time series that reflect BDM fluctuations over time. Because calculating BDM requires extensive precomputed data, neither the online OACC nor the standard standalone version was suitable for these large datasets. Consequently, the core routines of the R implementation of OACC were extracted and integrated directly into the processing pipeline for this analysis.

Binarization and BDM computations on such voluminous data are computationally demanding. To address this, E2C instances on Amazon Web Services were employed. Given a sampling rate of 400 Hz and the chosen block size (e.g., 16×16), each 10-minute recording yields 15,000 matrices ($400 \text{ Hz} * 600 \text{ seconds} / 16$). After computing BDM for every block, the sequence can be plotted as a time series with 15,000 time steps, showing the temporal dynamics of a patient's BDM profile. A full account of the processing steps is provided in [291, 292].

Algorithmic complexity (AC) for each block is aggregated along the temporal axis to produce time series that represent the evolution of AC across space and time. To enable these calculations, the essential two-dimensional component from the relevant software package was isolated and embedded into a controller program that manages data flow for this study. Raw biological signal datasets were prepared for OACC, and AC was estimated using a mapping procedure summarized in Tables 6.6–6.8.

Table 6.6. Unprocessed EEG data (Reproduced from the author's work [39])

Ch1	Ch2	Ch2	Ch4	Ch5	Ch6	Ch7	Ch8-Ch15	Ch16
-38	-13	9	-15	7	6	24	...	25
-45	-5	-2	-12	2	-1	17	...	21
-46	-11	-3	-14	2	2	16	...	17
-36	-21	-5	-16	7	9	16	...	20
-22	-23	-3	-20	13	16	8	...	17
-14	-16	3	-30	19	18	0	...	2
-3	-6	22	-36	17	14	-9	...	-5
2	-1	35	-44	18	12	-7	...	-19
0	-6	36	-46	20	16	4	...	-25
6	-15	34	-33	22	12	3	...	-27
15	-25	33	-27	19	5	7	...	-32
25	-21	34	-26	12	0	9	...	-34
33	-19	36	-33	2	-11	15	...	-23
26	-15	39	-26	-9	-9	9	...	-19
16	1	32	-25	-8	-5	7	...	-18
7	16	35	-26	-5	1	5	...	-20

Note: The provided data represents the voltages of EEG signals recorded from 16 electrodes. Each column represents a specific electrode, and each row represents an individual observation. Data are reproduced from the author's paper [39].

Table 6.7. Data that has been binarized according to the mean threshold value per row
(Reproduced from the author's work [39])

Ch1	Ch2	Ch2	Ch4	Ch5	Ch6	Ch7	Ch8-Ch15	Ch16
0	0	1	0	0	0	1	...	1
0	0	0	0	0	0	1	...	1
0	0	0	0	0	0	1	...	1
0	0	0	0	0	1	1	...	1
0	0	0	0	1	1	0	...	1
0	0	0	0	1	1	0	...	0
0	0	1	0	1	1	0	...	0
0	0	1	0	1	1	0	...	0
0	0	1	0	1	1	0	...	0
0	0	1	0	1	1	0	...	0
1	0	1	0	1	0	0	...	0
1	0	1	0	1	0	1	...	0
1	0	1	0	0	0	1	...	0
1	0	1	0	0	0	1	...	0
1	0	1	0	0	0	0	...	0
0	1	1	0	0	0	0	...	0

Note: Data are reproduced from the author's paper [39].

Table 6.8. Algorithmic complexity calculated via BDM [39]

Algorithmic complexity
373.5134 bits

Note: Data are reproduced from the author's paper [39].

Figure 6.4 shows the temporal behavior of Kolmogorov-Chaitin complexity (KCC) over a 10-minute EEG interval as evaluated by the Block Decomposition Method (BDM).

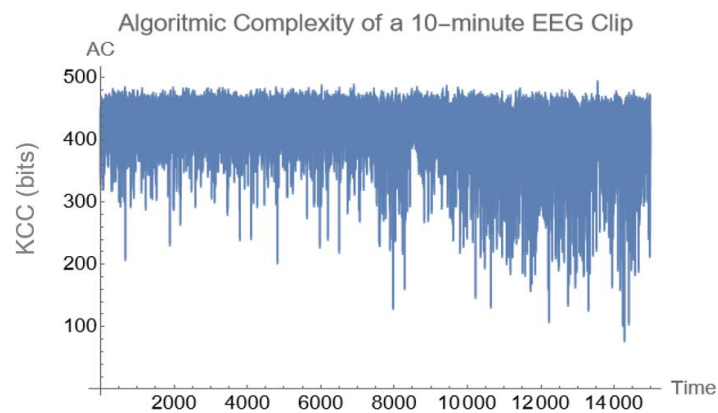


Figure 6.4. Temporal series describing KCC values for a 10-minute EEG segment.

Subsequently, the integer-rounded KCC values were mapped to letters (for example, 1 → "a", 2 → "b", 3 → "c", 4 → "d", etc.). This mapping produced text files whose lexicons encode algorithmic complexity. Further details of this encoding are provided in [292].

One tool applied at this stage is based on latent Dirichlet allocation (LDA). LDA views each document as a blend of topics and each topic as a blend of words, allowing texts to share themes rather than being rigidly separated. LDA concurrently estimates the topic distribution for each document and the word distribution for each topic. In our workflow, the KCC time-series files converted to text act as documents, with KCC values turned into letters that play the role of words. The R package "topicmodels" was used to fit an LDA model with two topics. From this model, per-topic-per-word probabilities ("beta") were extracted. Figure 6.5 displays the thirty most common tokens, i.e., the most frequent KCC values represented as words, associated with each topic, highlighting the dominant KCC values per topic.

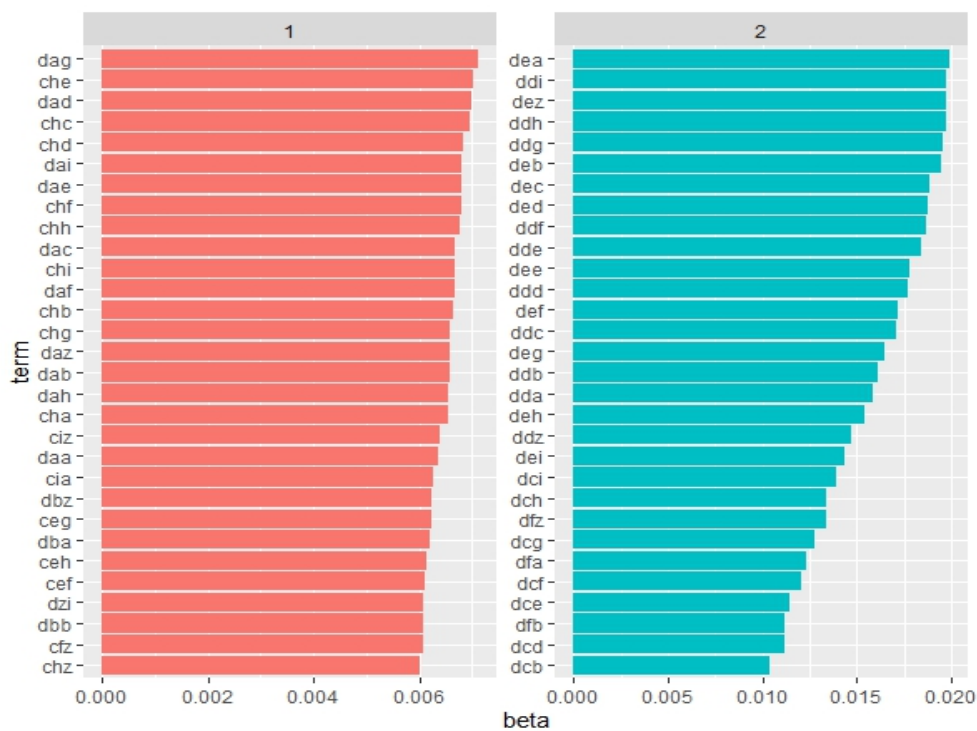


Figure 6.5. The predominant KCC values (converted to words) associated with every topic.
Reproduced from the author's work [292].

By translating these 'words' back to their numeric counterparts and plotting the distribution of the top 100 BDM values for each topic, the plots in Figure 6.6 were obtained.

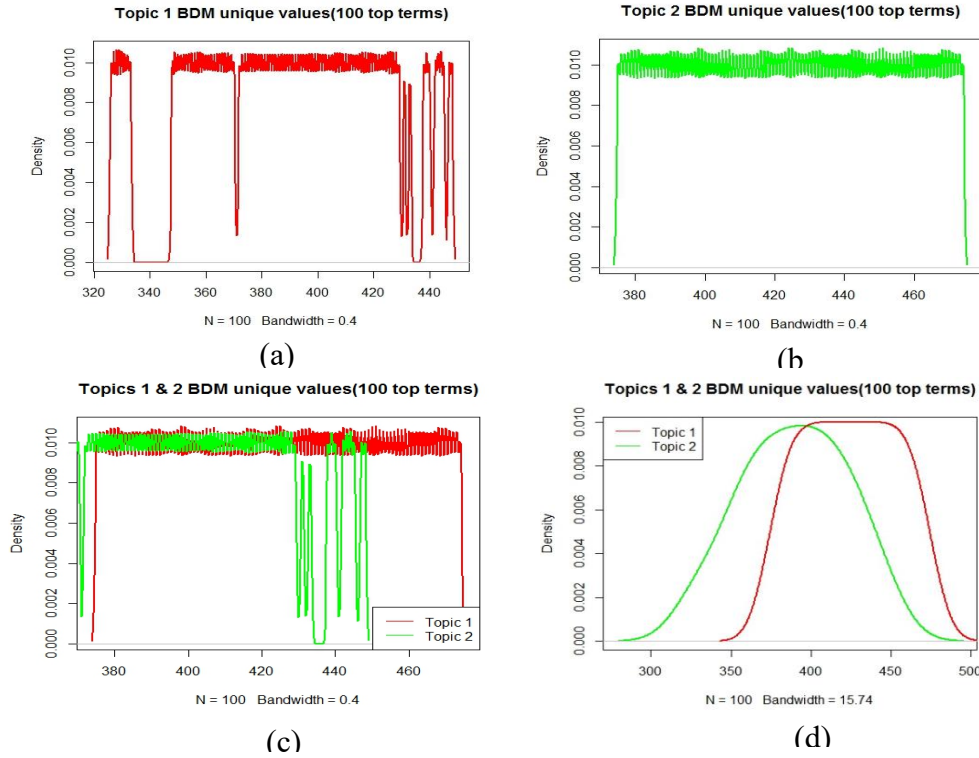


Figure 6.6. Distinct values for topic 1 and topic 2 BDM - upper graphs (a & b). The lower left graph (c) is the amalgamation of the higher graphs. The distinction between these graphs is plainly apparent. The plot in the bottom-right corner (d) illustrates that a similar superposition occurs with an increased bandwidth (15.74 compared to 0.4), highlighting the skewness more distinctly. [292]

In the LDA framework, observable variables are the words (here, KCC/BDM values encoded as letters), while topics and document-topic mixtures are latent variables. A sparse Dirichlet prior models the topic-word distributions, reflecting the expectation that within a topic only a small subset of words will have high probability.

The separation between "interictal" and "preictal" states is further visualized with boxplots (Figure 6.7). Even after trimming extreme KCC values, the "preictal" set exhibits instances where the high-probability words for the topics are well separated (highlighted in red frames), whereas in the "interictal" set these high-probability words overlap (indicated by a green border).

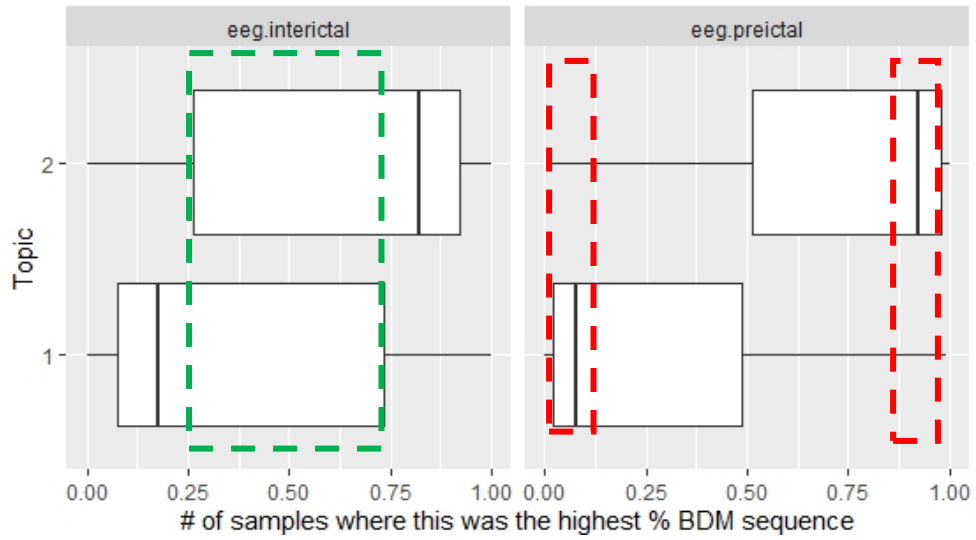


Figure 6.7. The distinction between the topics illustrated by box plots. Reproduced from the author's work [292].

In the LDA setting, words (numeric KCC values converted to letters) are observed variables and the remaining quantities are latent. The sparse Dirichlet prior assumes only a few words per topic carry most of the probability mass, yielding an asymmetric word-distribution within topics.

The "preictal" and "interictal" labels correspond to particular collections of documents; each document aligns with one entry in the AC time series derived from EEG segments. A "KCC value" is an integer-to-letter mapping representing specific tokens. Each text document is mapped to an array template with 1,100 slots corresponding to permutations of 10 characters arranged as groups of three and two; theoretically the dictionary can contain up to 1,100 entries. In practice, a document converted from an AC time series typically contains 400–500 unique tokens, reflecting the number of distinct AC values represented as words. Each position in the template array denotes a specific word's beta-log ratio.

The transformed data, prepared as described, were then used for machine learning. The final dataset structure (see Table 7.1, section 7.2.2) contains 1,100 columns holding beta-log ratio values (V1–V1100) plus a label column (0 = normal, 1 = preictal clip) [292].

This representation demonstrates one viable way to leverage EEG processed with algorithmic dynamics and NLP tools for machine-learning tasks. Other approaches that combine algorithmic-complexity preprocessing with NLP techniques are explored in the following sections.

6.2.3. Assessment of the Kolmogorov-Chaitin complexity regarding sepsis dataset

A comparable methodology was applied to the sepsis dataset [317]. To compute AC via BDM, six selected physiological parameters were arranged into 3×3 matrices that encapsulate

the mean values over three consecutive hours. Table 6.9 gives an example 3×3 block containing HR, SBP, and DBP. The mapping used to compute AC from the raw physiological signals is summarized in Tables 6.9–6.11.

Table 6.9. Preliminary sepsis data (HR, SBP, and DBP), organized in a 3×3 array [317]

HR	88	83	74.5
SBP	95.5	100.25	90.25
DBP	51	52	50.5

Note: HR - heart rate, SBP - systolic blood pressure, and DBP - diastolic blood pressure. Data in tables 6.9 to 6.11 are reproduced from the author's paper [317].

Table 6.10. Binarized information (employing the threshold value on a per-row basis) [317]

HR	1	1	0
SBP	1	1	0
DBP	0	1	0

Table 6.11. Algorithmic complexity derived from BDM [317]

Algorithmic complexity
17.4911 bits

Reference [293] provides a more detailed description of a similar mapping. Six physiological markers were selected for modeling. The final dataset layout (Table 6.12) includes 15 columns: two columns for KCC/BDM values and twelve columns capturing hour-to-hour differences for the six parameters, plus a label. The example below shows five clinical cases.

Table 6.12. The format of the sepsis data used for machine learning

AC1	AC2	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	Label
17.213	17.166	5	3.5	-0.1	12	4	0	6	-2	-0.05	-31.5	-7.5	3.5	0
18.980	17.232	0.5	0.5	-0.05	9	9.5	-3.5	-5	-1	-0.25	0	-13.5	4.5	0
15.942	17.019	8	0	-0.1	8	3	-1.17	-9	0	0	-8	-8	-1.16	0
15.601	16.162	3	0	-0.06	6.5	3	1.5	-1.67	-2	0.03	-4.7	-1.7	0	1
16.476	16.956	-4	-1	-0.7	-2	-2	0	-5.5	1	-0.2	-11	-21.5	-5.5	1

Note: AC1 and AC2 represent the KCC. V1 to V12 indicate the variations in the values of six parameters (HR, O₂Sat, Temp, SBP, DBP, and Resp) between three consecutive hours. The label "0" corresponds to non-sepsis, while the label "1" corresponds to sepsis.

6.3. Utilization of topological representation learning techniques for research data

6.3.1. Text representation learning. Word embedding

Learning representations for text is essential across many domains, motivating the use of extensive collections of written materials—books and other texts—for downstream tasks. Earlier approaches relied on statistical measures like TF-IDF. However, advances in computing power and digital data availability have driven the rapid development of natural language processing (NLP), which is now widely applied in fields ranging from biology and chemistry to medicine and commerce.

To process textual content computationally, raw text (the corpus) must be converted into machine-friendly formats. A primary preprocessing step is tokenization, which isolates normalized textual units—typically words—into discrete, cleaned tokens suitable for modeling. Concretely, tokenization converts a noisy string into a sequence of tokens; for example, "natural language processing" → ['natural', 'language', 'processing'].

Furthermore, words can be segmented into smaller significant units and amalgamated to create pairs, triplets, quadruplets, etc., known as n-grams. An further preprocessing method is stemming, which includes eliminating the subtle meaning variations introduced by plural forms, possessive suffixes, or various verb conjugations. For instance, 'house' serves as the root for both 'housing' and 'houses.' Lemmatization, analogous to stemming in some instances, entails consolidating many words with comparable meanings, notwithstanding possible spelling variations (e.g., the lemma for 'runs,' 'running,' and 'ran' is 'run').

Words may also be split into smaller meaningful subunits and recombined into n-grams (bigrams, trigrams, etc.). Additional preprocessing includes stemming—trimming morphological variations like plurals, possessives, or verb conjugations (e.g., 'housing' and 'houses' reduced to 'hous')—and lemmatization, which groups inflected forms under their canonical lemma (e.g., 'runs', 'running', 'ran' → 'run'). While both reduce dimensionality, lemmatization often preserves semantics more accurately by considering word meaning. Tools such as the NLTK stemmers (Snowball, Porter, Lancaster) and the WordNet Lemmatizer implement these methods.

Applying these operations (notably tokenization) yields a vocabulary: the ordered set of unique tokens in a corpus. From this vocabulary, words can be represented numerically as one-hot vectors. Arranging one-hot vectors in sequence reconstructs the original text in numerical form, facilitating computation and subsequent processing.

Vocabulary normalization - standardizing tokens by conflating variants with similar meaning - is crucial. This may use stemming, lemmatization, lowercasing, and TF-IDF weighting as part of the normalization pipeline.

Term frequency-inverse document frequency (TF-IDF)

TF-IDF is the numerical statistic that shows the relevance of a phrase in the document corpus and it is still the fundamental technique of text classification. It combines the term's frequency within a document with how rare the term is across all documents, producing a more informative feature for classification. Let $w(t_i, d)$ be the weight of word t_i in document d , $tf(t_i, d)$ the frequency of t_i in d , N the total number of documents and n_{ti} the number of documents containing t_i ; TF-IDF weights are computed accordingly:

$$w(t_i, d) = \frac{tf(t_i, d) \times idf(t_i)}{\sqrt{\sum_{t_i \in d} [tf(t_i, d) \times idf(t_i)]^2}}, \quad (6.15)$$

$$idf(t_i) = \log(N/n_{ti}) + 1. \quad (6.16)$$

Word embedding

Word embedding refers to representing words as numeric vectors in a high-dimensional space such that semantically similar words are positioned closer together. Building on work by Bengio, Hinton, and colleagues [295, 296], most modern embedding approaches since 2005 use neural-network-based models rather than purely probabilistic or algebraic formulations. In NLP tasks, terms from sentences or documents typically serve as features.

Two common vector schemes are one-hot encoding and distributed (dense) representations. One-hot vectors have dimensionality equal to vocabulary size, with a single one indicating the word's index and zeros elsewhere; they suffer from very high dimensionality, sparsity, and the inability to capture semantic relationships. Hinton (1986) introduced distributed representations [297], assigning low-dimensional dense real-valued vectors to words and enabling semantic similarity to be measured via distances in vector space.

Bengio et al. [298] integrated distributed word representations into a neural language model (NNLM) consisting of input, mapping, hidden, and output layers.

Word2vec model

Mikolov et al. [299] improved training efficiency by removing NNLM's nonlinear hidden layer, producing the Word2vec family of models—Continuous Bag-of-Words (CBOW) and Skip-gram (SG). Both are three-layer architectures (input, projection, output). CBOW predicts a target word given its context, while Skip-gram predicts context words given a target word with the following mathematical formalism behind the process:

$$context(w_k) = \{w_{k-t}, w_{k-(t-1)}, \dots, w_{k+(t-1)}, w_{k+t}\}, \quad (6.17)$$

$$\text{maximize } \sum_{(w,k) \in D} \log P(k|w), \quad (6.18)$$

$$P(k|w) = \exp(v_k \cdot v_w) / \sum_{\{k' \in V\}} \exp(v_{\{k'\}} \cdot v_w), \quad (6.19)$$

$$P(w|k) = \exp(v_w \cdot v_k) / \sum_{\{w' \in V\}} \exp(v_{\{w'\}} \cdot v_k). \quad (6.20)$$

Figure 6.8 illustrates the CBOW and Skip-gram architectures [300].

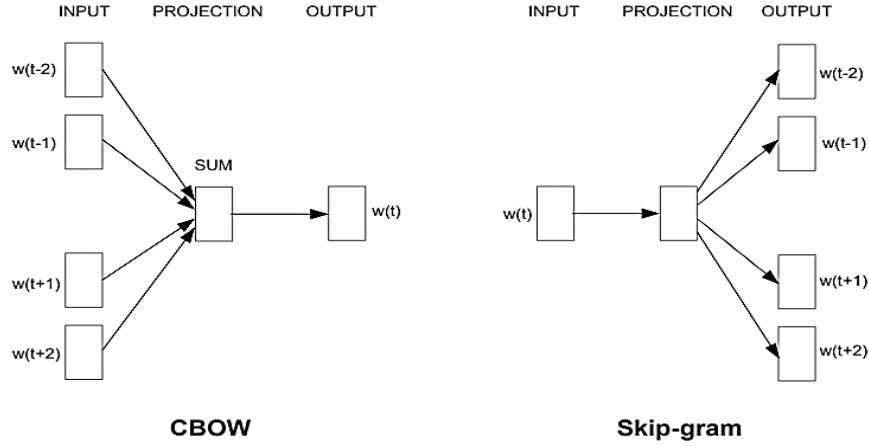


Figure 6.8 The Continuous Bag of Words & Skip-gram Models.

GloVe Model

GloVe (Global Vectors) combines global count-based methods (e.g., matrix factorization like PCA) with local prediction-based approaches (like Word2vec). Unlike Word2vec's feedforward neural formulation, GloVe constructs embeddings by factorizing a global word co-occurrence matrix, leveraging ratios of co-occurrence probabilities to capture semantic relationships. The model learns to fit the dot product of word vectors to the log of co-occurrence counts and the mathematical formalism behind it as follows [301]:

$$w_i^T + \tilde{w}_k + b_i + \tilde{b}_k = \log(X_{ik}), \quad (6.21)$$

$$f(X_{ik}) = \begin{cases} \left(\frac{X_{ik}}{x_{max}}\right)^\alpha, & \text{if } X_{ik} < x_{max}, \\ 1, & \text{otherwise} \end{cases} \quad (6.22)$$

where w is the vector representation of a word, $\tilde{w}_k$ is the vector representation of the context word. The biases, b_i and $\tilde{b}_k$, are for the i -word and k -word contexts, respectively, the co-occurrence matrix X and weight function $f(X_{ik})$.

The cost function can be presented as [301]:

$$J = \sum_{i,k=1}^v f(X_{ik})(w_i^T \tilde{w}_k + b_i + \tilde{b}_k - \log(X_{ik}))^2. \quad (6.23)$$

Because implementing GloVe from scratch was impractical for this project, GloVe was not used on these datasets.

fastText model

fastText, developed by the Word2vec team [302], has an architecture similar to CBOW but includes subword information and a hidden layer between input and output (see Figure 6.9) [303]. Words are decomposed into character n-grams and represented as the sum of n-gram vectors; this subword modeling aids representation of short words and morphological variants. The resulting word vector is then trained within a CBOW-like objective. Such subword-aware models are particularly valuable for languages with large vocabularies and many low-frequency words.

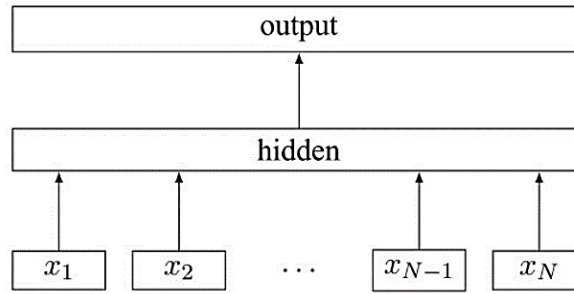


Figure 6.9. The architectural framework of fastText for a sentence that has n-gram attributes.

BERT model

BERT stands for Bidirectional Encoder Representations from Transformers. It is a pretraining technique that combines masked language modeling and bidirectional transformer encoders[305]. Unlike traditional unidirectional models, BERT learns context from both left and right by predicting masked tokens in a sequence, yielding powerful contextualized representations that perform well across many NLP tasks with minimal architecture changes.

Long Short-Term Memory (LSTM) Model

Standard feedforward networks struggle with sequence learning because they do not capture dependencies across time. Recurrent neural networks (RNNs) address this by connecting hidden states over time to learn sequential structure. However, vanilla RNNs have difficulty with long-range dependencies due to vanishing gradients. LSTM networks mitigate this with gated mechanisms—input gate i , forget gate f , output gate o , and a memory cell. Figure 6.10 shows the

LSTM architecture [306, 307]. The forget gate f controls what to drop from the memory cell; its output ranges between 0 and 1 and is computed as follows:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + V_f c_{t-1} + b_f). \quad (6.24)$$

Let x_t and h_{t-1} denote inputs, with weights W_f and U_f for input and recurrent connections respectively. c_{t-1} is the previous cell state, V_f links c_{t-1} to the forget gate f , b_f is the bias, and $\sigma(\cdot)$ is the sigmoid activation.

The input gate identifies the precise information which serves for updating in the memory cell using the tangent activation function $\tanh(\cdot)$:

$$\begin{cases} i_t = \sigma(W_i x_t + U_i h_{t-1} + V_i c_{t-1} + b_i) \\ c_in_t = \tanh(W_c x_t + U_c h_{t-1} + V_c c_{t-1} + b_c). \end{cases} \quad (6.25)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot c_in_t$$

The output gate o governs the cell's output value. Its computation involves weights linking x_t , h_{t-1} , c_{t-1} , and the bias terms:

$$\begin{cases} \sigma_t = \sigma(W_o x_t + U_o h_{t-1} + V_o c_{t-1} + b_o) \\ h_t = o_t \cdot \tanh(c_t) \end{cases} \quad (6.26)$$

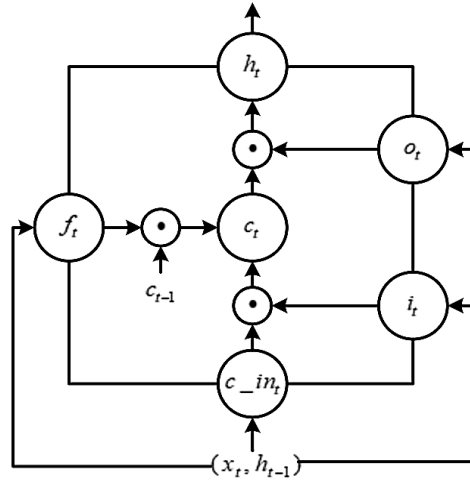


Figure 6.10. Diagrammatic depiction of the configuration of a Long Short-Term Memory (LSTM) network (adapted from [199]).

In both conventional RNNs and standard LSTMs, information flows unidirectionally; the state at time t depends only on previous inputs. Two-way LSTM (BiLSTM) architectures combine forward and backward LSTM layers to record more rich contextual information at each time step. This lets them take into account both past and future context [307].

The BiLSTM assigns equal capacity to inputs across the sequence; salient tokens determine the classification outcome. In this workflow, word vectors are derived via a sequence-to-sequence process, and the weighted word vectors feed into a BiLSTM. The BiLSTM outputs serve as

document representations, which are passed to a feedforward classifier to produce the final label (interictal vs. preictal).

6.3.2. Representation learning and application of NLP tools to transformed (textual) EEG data

Utilizing representation learning techniques for numerical EEG data

Representation learning for numerical EEG data was first applied to the arithmetic-test dataset. The following illustrations show the use of basic representation-learning techniques on numerical time series, specifically the AC (algorithmic complexity) sequences derived from the arithmetic task. Subsequently, NLP tools are applied to the textual representations obtained from the epileptic EEG dataset. The suite of models tested includes Word2Vec, fastText, BiLSTM, and BERT, as described previously.

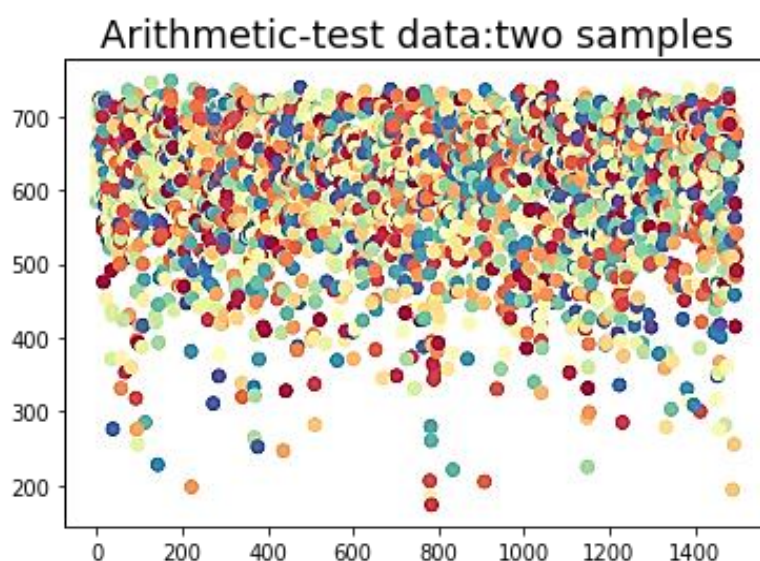
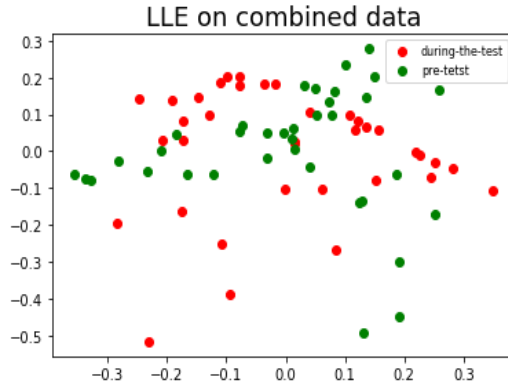
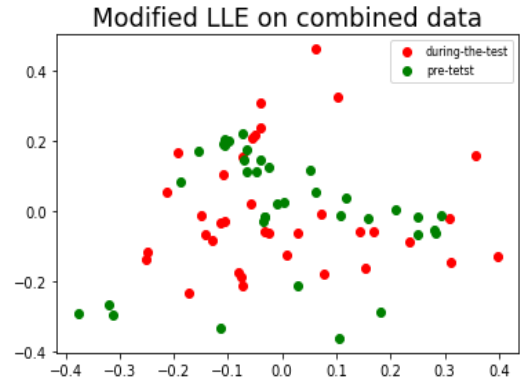


Figure 6.11. A scatter diagram representing two samples extracted from the arithmetic examination dataset.

Figure 6.11 displays a scatter plot of two samples from the arithmetic dataset, where the x-axis is time and the y-axis is algorithmic complexity; a spectral colormap is used. The results of the nonlinear embedding methods (Figures 6.12–6.14) are shown next: green indicates pre-test algorithmic complexity and red indicates complexity during the test.

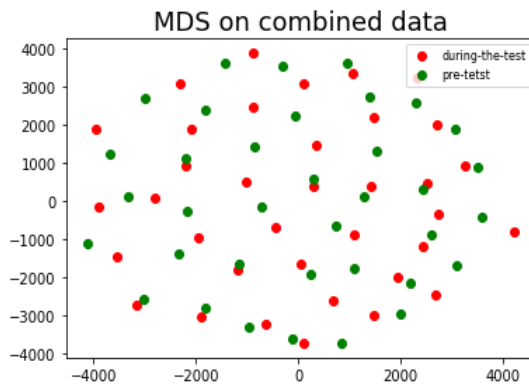


(a)

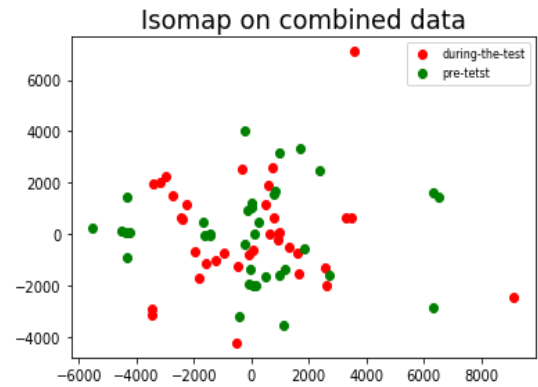


(b)

Figure 6.12. Locally Linear Embedding (LLE) applied to arithmetic-test data — (a) original LLE and (b) a modified LLE.

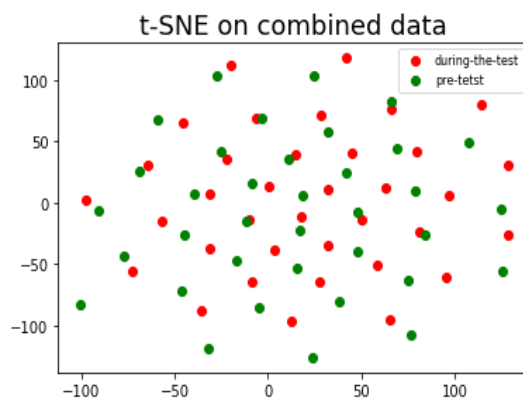


(a)

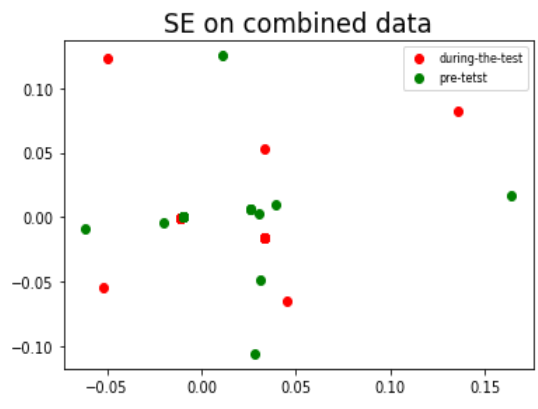


(b)

Figure 6.13. Multidimensional Scaling (MDS) (a) and Isomap (b) on arithmetic-test data.



(a)



(b)

Figure 6.14. t-SNE (a) and spectral embedding (b) on arithmetic-test data.

The nonlinear methods produce embeddings with less overlap than the raw representation (compare Figure 6.11 to Figures 6.12–6.14), which can help improve classification performance [308].

Word2Vec model

Word2Vec was trained on the epileptic EEG-derived text in both standard variants using the Gensim library [309] and the H2O Word2Vec implementation. Because the preictal and interictal sets differed in size, additional preictal segments were generated to balance the data: 439 preictal segments for Dog 1 and 480 for Dog 2. The datasets were then partitioned into pretraining, training, and test subsets as summarized in Table 6.13.

Table 6.13. Datasets used to build Word2Vec models*

The source of the data set	Set for pretraining / building the word embedding model	Set for training/ building of a GBM classifier	Test set
Dog 1	938 “documents”/clips (14,070,000 “words”)	434 “documents”/clips: • preictal – 219 • interictal- 215 (6,510,000 “words” in total)	69 “documents”/clips: • preictal – 24 • interictal- 45 (1,035,000 “words” in total)
Dog 2	1,464 “documents”/clips (21,960,000 “words”)	465 “documents”/clips: • preictal – 240 • interictal- 225 (6,975,000 “words” in total)	92 “documents”/clips: • preictal – 42 • interictal- 50 (1,380,000 “words” in total)

*The same splits were used for fastText and BiLSTM models.

Separate word-embedding models were trained for each dog. After tokenizing the training data (using the tokenizer developed earlier), a GBM classifier was trained to distinguish preictal from interictal segments and then evaluated on the held-out test sets.

Uniform Manifold Approximation and Projection (UMAP) [310] diagnostics (Figure 6.15) were used to inspect embeddings at both global (via PCA coloring) and local scales. The approach consisted of computing a three-dimensional PCA projection to capture global structure, mapping the 3D PCA coordinates into RGB colors, and then coloring the UMAP embedding with these PCA-derived hues to visualize how the global layout is preserved. Local dimensionality was also examined to assess regional embedding fidelity: regions where points show uniformly low local dimensionality are considered more reliable.

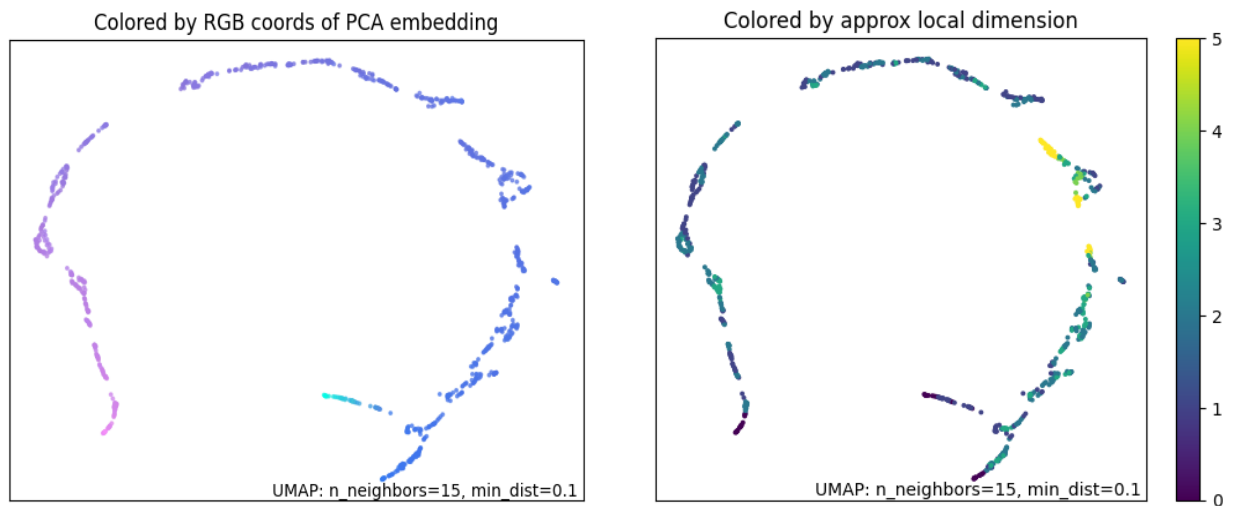


Figure 6.15. UMAP diagnostic visualizations for Dog 2 - left: PCA-based coloring showing preservation of global structure; right: local-dimension view highlighting areas of reduced embedding precision (yellow regions)..

fastText

fastText enables fast training on large corpora and generates vectors for out-of-vocabulary words via subword n-grams. However, the EEG-AC “language” consists of fixed tokens, so the typical advantages of n-grams, stemming, and lemmatization are rarely applicable here. Figure 6.16 shows a two-component PCA of 30 AC terms encoded by fastText vectors.

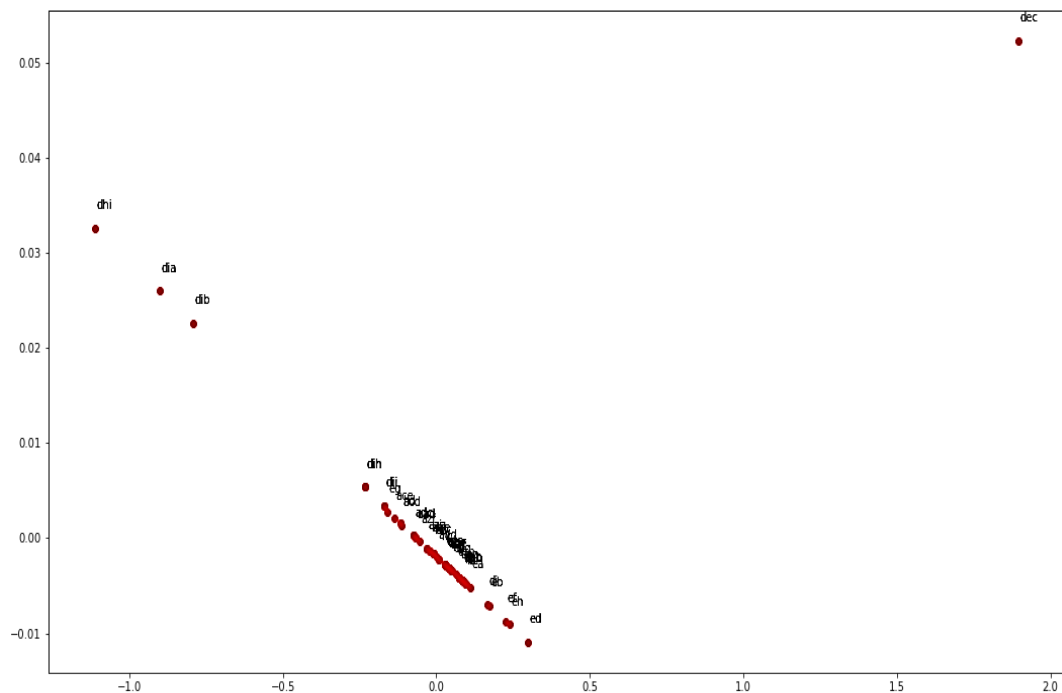


Figure 6.16. PCA of 30 algorithmic-complexity terms (two principal components).

In Figure 6.16, the upper, line-like cluster corresponds to high-AC tokens (e.g., "dhi" 489 bits, "dia" 491 bits, "dib" 492 bits), while the lower cluster corresponds to low-AC tokens (e.g., "eb" 52 bits, "eh" 58 bits, "ed" 54 bits).

BERT model

Because the AC-based “brain complexity language” is artificial and unlike natural text, no suitable pretrained BERT model could be reused. Therefore, several BERT models were trained from scratch on the prepared data. The BERT variants used here are configured for shorter sequences (<512 tokens), whereas Word2Vec-based models were capable of much longer sequences (up to ~15,000 tokens) corresponding to a full 10-minute EEG segment transformed into AC terms. Consequently, the data splits for BERT differ from those used for Word2Vec; Table 6.14 summarizes the BERT datasets.

Table 6.14. Data sets that were utilized in constructing the BERT models

The source of the data set	Set for pretraining / building the word embedding model	Set for fine-tuning	Test set
Dog 1	28,069 “documents”/sentences	12,995 “documents”/sentences: • preictal – 6.557 • interictal- 6.438	2,036 “documents”/sentences: • preictal – 718 • interictal- 1.318
Dog 2	43,920 “documents”/sentences	13,933 “documents”/sentences: • preictal – 7192 • interictal - 6741	2,754 “documents”/sentences: • preictal – 1.257 • interictal- 1.497

BiLSTM model

The BiLSTM-based classifier used a feedforward network with ReLU activations, a dropout rate of 0.2, and the Adam optimizer with a learning rate of 3e-4. Figure 6.17 presents the BiLSTM classification architecture: the left panel shows feature extraction from text, and the right panel shows the classification/regression head. NodeNum denotes the number of units in the LSTM hidden layer.

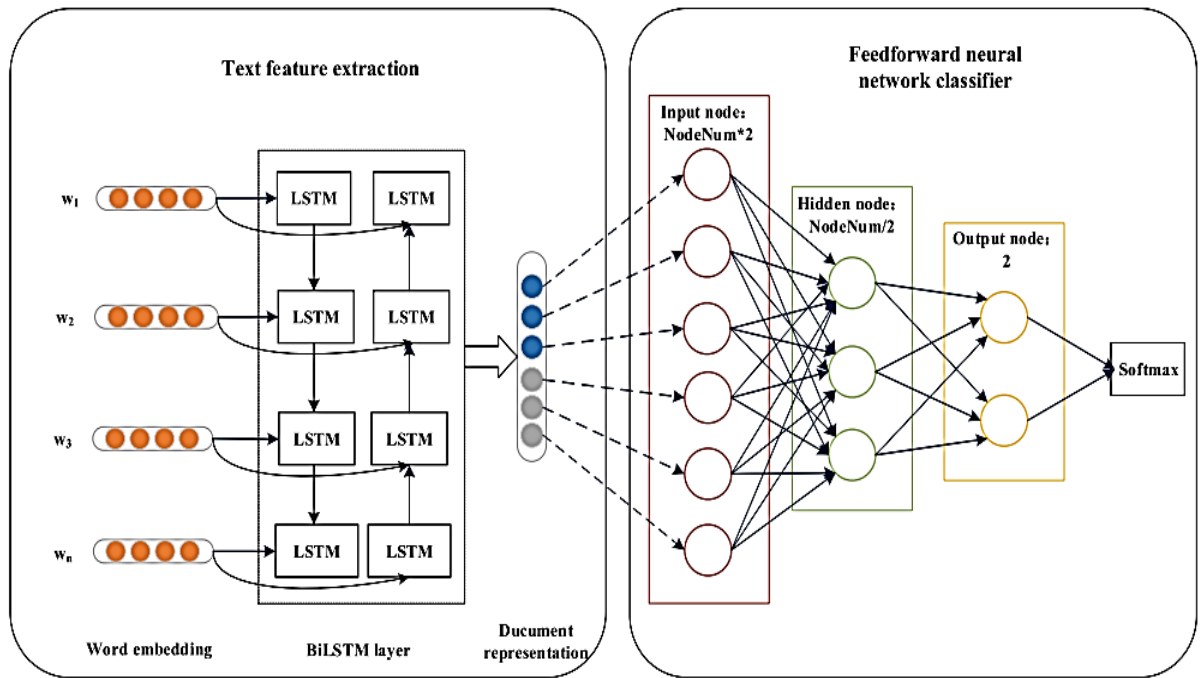


Diagram 6.17. BiLSTM-based classifier schematic.

6.3.3. Representation learning on sepsis data

Linear methods: PCA and LDA

When high-dimensional data must be visualized or summarized, linear dimensionality-reduction methods like principal component analysis (PCA) and linear discriminant analysis (LDA) can be used. PCA and LDA are among the most fundamental techniques for generating lower-dimensional representations that capture structure in the data [311–313].

Principal component analysis

PCA finds a lower-dimensional representation that captures maximal variance. Suppose we have n observations in a p dimensional space. PCA finds the linear combinations of the original features (principal components) that explain most of the variability in the data. The loading vector of the first principal component defines the direction of greatest variance. Plotting the principal components provides a compact view of the dataset structure (Figure 6.18).

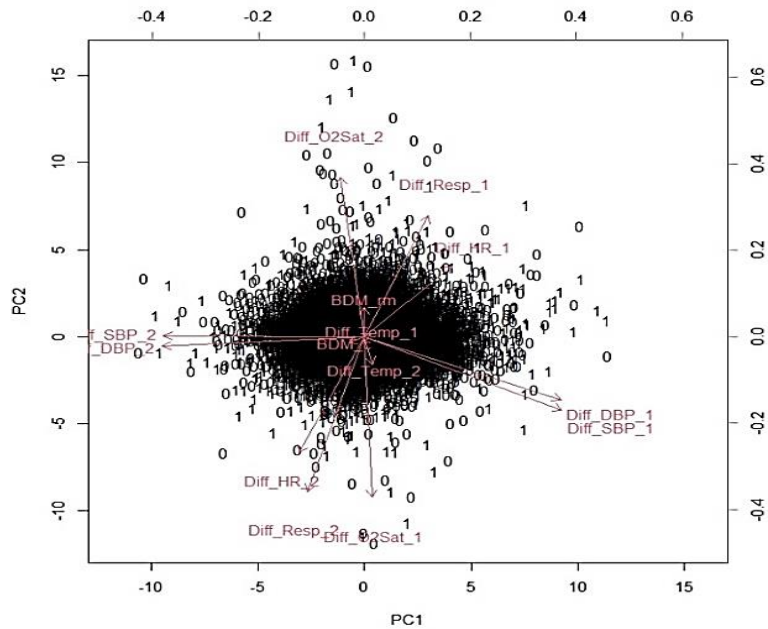


Figure 6.18. The biplot of the first two principal components.

The first principal component is driven largely by systolic and diastolic blood pressure variability; the second is influenced by oxygen saturation and respiratory rate. Labels: 0 = non-sepsis, 1 = sepsis.

Another interpretation of the first principal component is as the line in p -dimensional space minimizing mean squared distance from the observations. The scree plot and cumulative variance (Figure 6.19) show explained variance per component: PC1 explains 18.40%, PC2 explains 10.92%, and eleven components together explain 90% of variance.

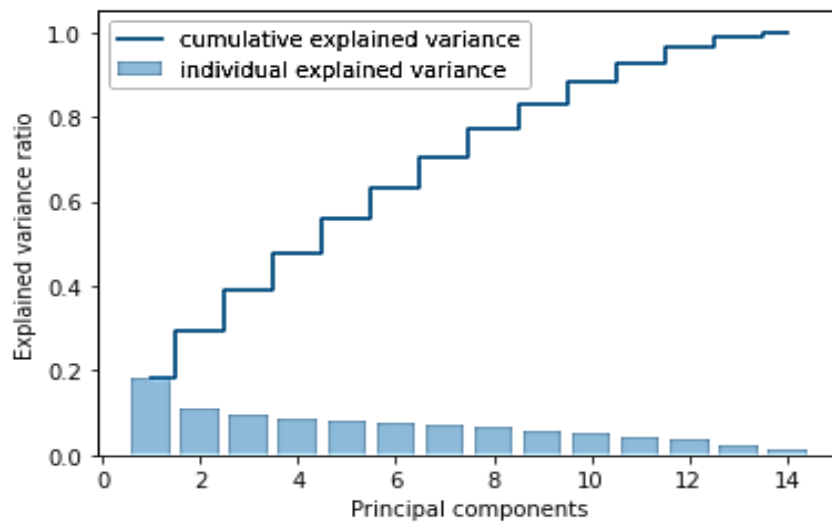


Figure 6.19. Individual and cumulative explained variance for the sepsis dataset.

The first principal component accounts for 18.40% of the variability in the data, followed by the subsequent principal component, which accounts for 10.92% of the variability. Eleven principal components collectively account for 90% of the variability.

PCA results depend on feature scaling—lack of standardization can skew results when variables have different units. How many principal components to keep depends on the purpose of the analysis and heuristics such as the elbow technique can help. Using reduced PCA representations ($n \times k$ matrices) for downstream tasks such as classification often improves robustness, since informative signal tends to concentrate in leading components. PCA-transformed sepsis data with reattached labels were used to train ML classifiers; performance is reported in Table 6.15.

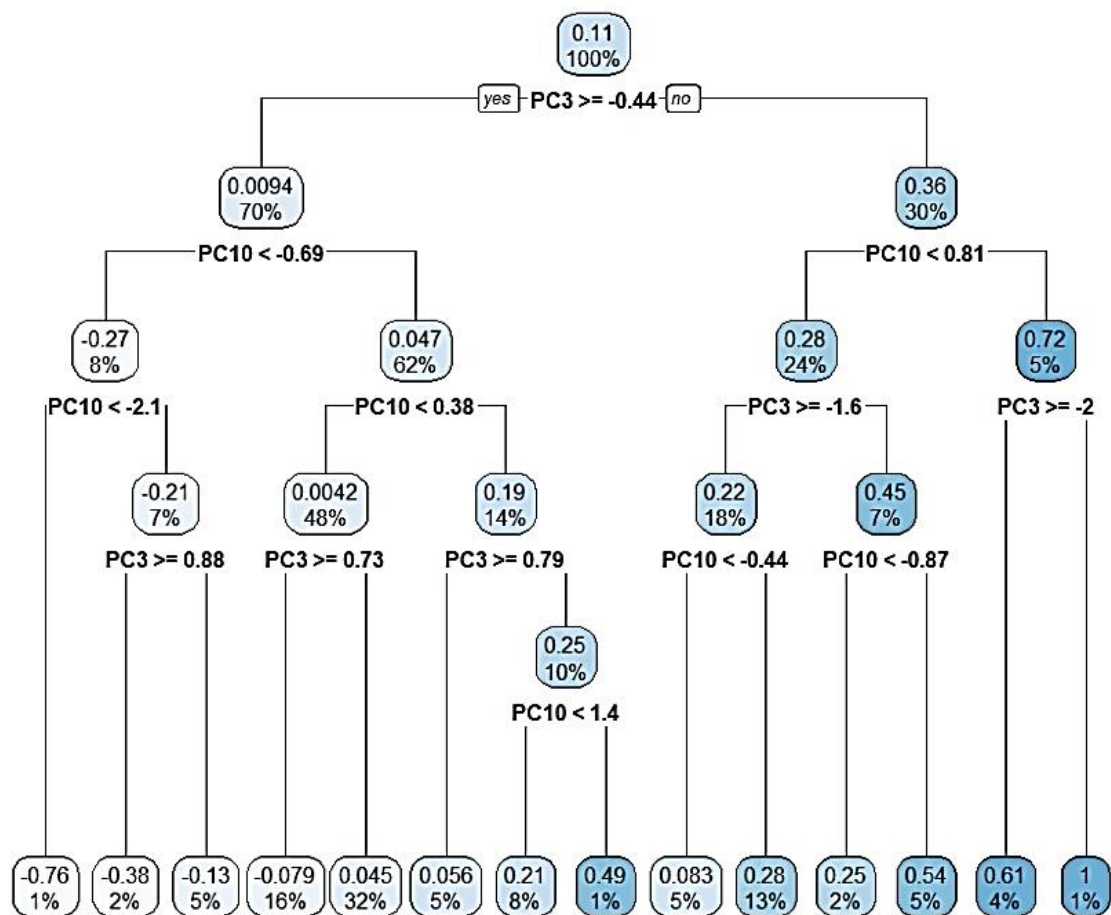


Figure 6.20. Example decision tree predicting patient body temperature using principal components 3 and 10.

Table 6.15. Performance of a GBM classifier on PCA-transformed data after re-attaching original labels

ML Performance metric	PCA data with 11 principal components	PCA data with 14 principal components
Accuracy	0.700	0.711
Recall	0.770	0.774
Specificity	0.663	0.677
AUC	0.789	0.801

Standard PCA is a linear, unsupervised, global feature extractor. Nonlinear extensions exist, such as kernel PCA (KPCA) [314] and probabilistic PCA (PPCA) [315].

Linear Discriminant Analysis

LDA is a supervised linear projection that aims to find the maximum class separation by reducing the within-class dispersion and maximizing the between-class scatter. In contrast to PCA, which identifies directions of maximal variance irrespective of labels, LDA optimizes discrimination between classes. Applied to the binary sepsis problem, LDA yields a single discriminant. Using two features (temperature derivatives), the LDA coefficients were computed as -2.499792 and -2.151678 . The decision threshold c (commonly 0.5) classifies instances: scores $> c$ are labeled sepsis (1), otherwise non-sepsis (0):

$$-2.499792 \frac{dT_{\text{temperature}_1}}{dt} - 2.151678 \frac{dT_{\text{temperature}_2}}{dt} > c. \quad (6.27)$$

Figure 6.21 shows the LDA projections for the training set (90% of data).

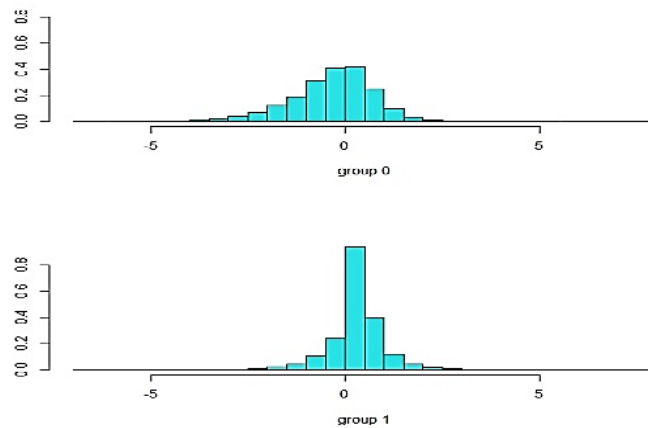


Figure 6.21. LDA plots for the sepsis training set using two features (three-hour derivatives of body temperature); top: non-sepsis class, bottom: sepsis class.

When all 14 features are used, the discriminant expands to include coefficients for each feature. LDA results on the 10% held-out test set are summarized in confusion matrices (Table 6.16).

Table 6.16. Confusion matrices for LDA on the sepsis test set

LDA.class	0	1
0	1629	850
1	157	124

LDA.class	0	1
0	1629	850
1	189	169

Note: Left: LDA with two features; Right: LDA with all 14 features.

Figure 6.22 depicts a visual overview of the application of PCA and LDA techniques to the sepsis dataset.

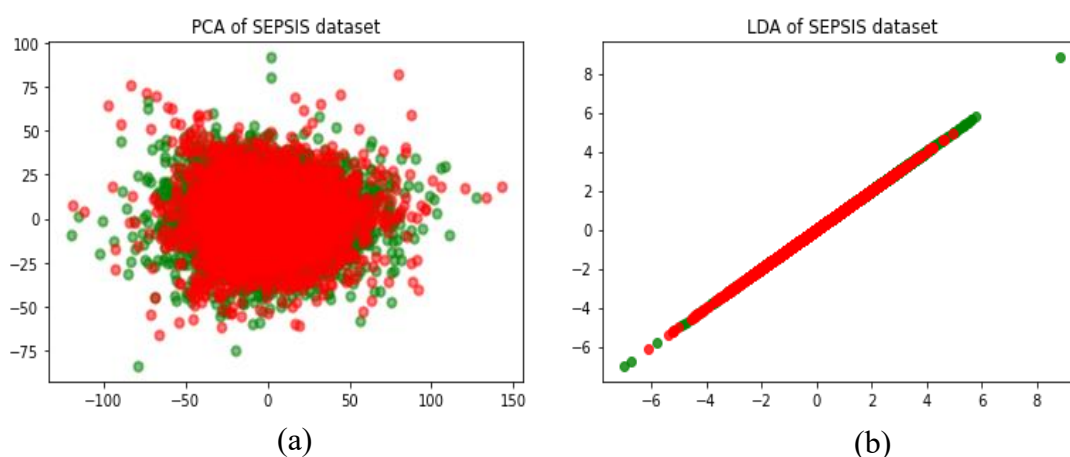


Figure 6.22. PCA and LDA plots of the sepsis dataset (first two principal components on axes); sepsis cases in red, non-sepsis in green.

From the plots, linear separation appears unlikely. PCA focuses on variance-based representation and dimensionality reduction without using labels, while LDA optimizes projection to maximize separability between classes.

Non-linear representation learning

Non-linear representation learning captures complex, nonlinear relationships among features. These approaches can be unsupervised (learning structure from unlabeled data) or semi-supervised (leveraging unlabeled data to extract features while pairing inputs with labels for supervised learning). Unsupervised feature learning aims to find low-dimensional features summarizing the structure of high-dimensional inputs.

Figure 6.23 visualizes three of the fourteen sepsis features in 3D with a spectral colormap.

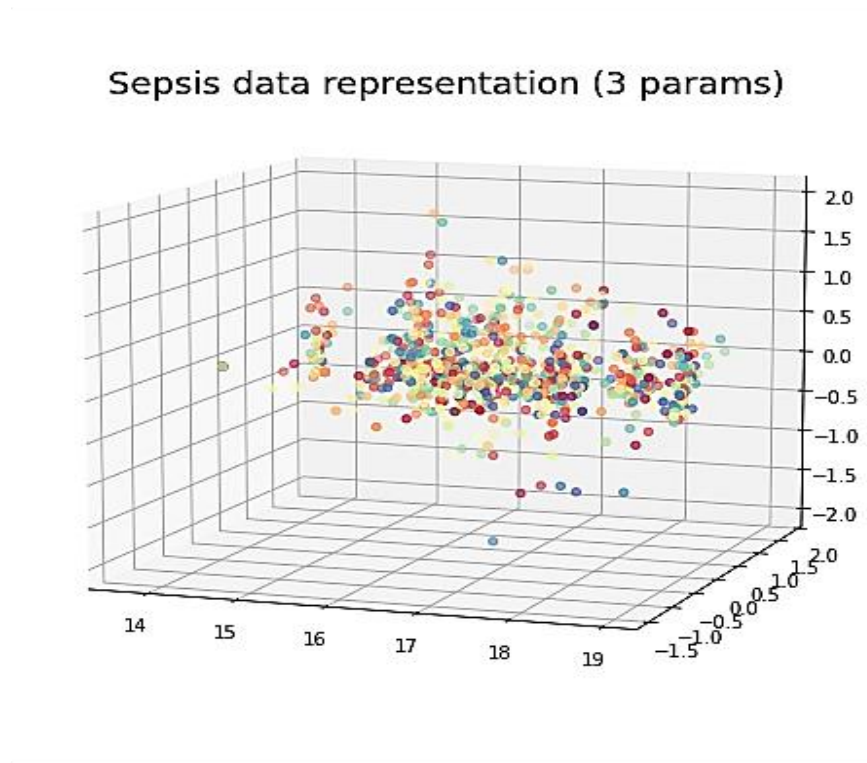


Figure 6.23. 3D representation of three sepsis variables.

Figures 6.24–6.26 compare several standard nonlinear embeddings applied to the sepsis dataset (red = sepsis, green = non-sepsis) and the arithmetic-test dataset (green = pre-test, red = during-test):

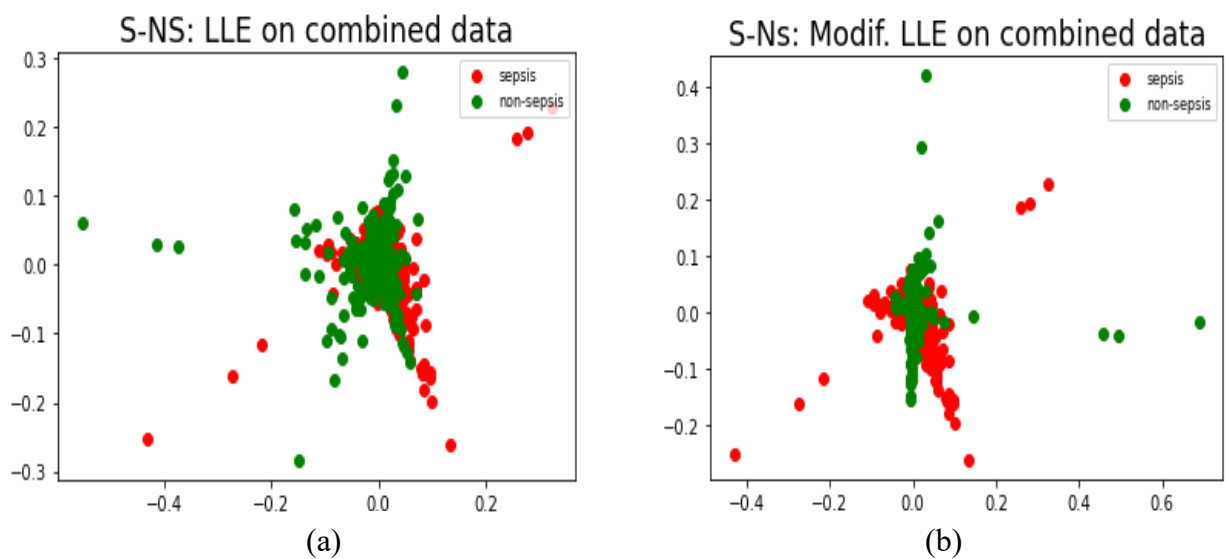


Figure 6.24. LLE (a) and modified LLE (b) on sepsis data.

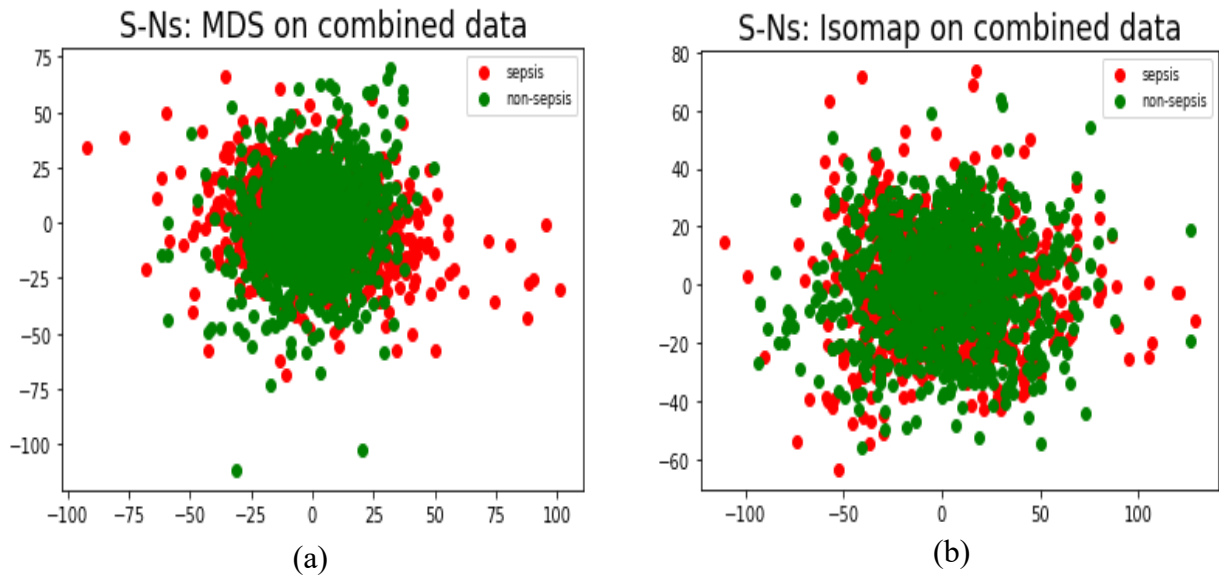


Figure 6.25. MDS (a) and Isomap (b) on sepsis data.

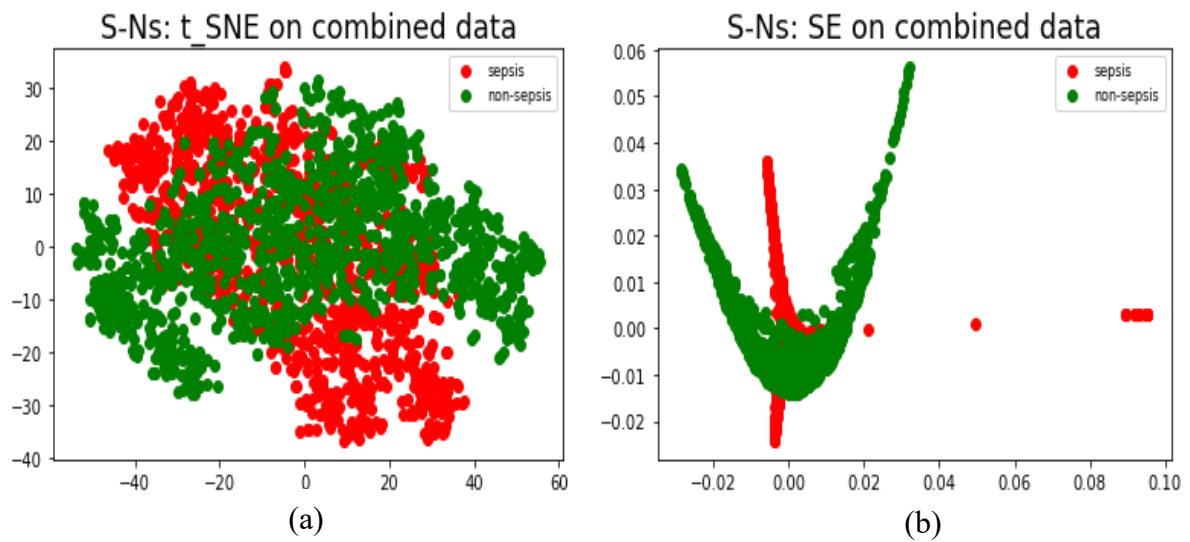


Figure 6.26. t-SNE (a) and spectral embedding (b) on sepsis data.

6.4. Intermediary results and stage concluding remarks

Sections 6.1.1 and 6.1.2 discussed the prevalence of missing values in the sepsis dataset and multiple imputation strategies. This section evaluates how different imputation methods affect downstream ML model performance.

After imputation, sepsis data were split into training (85–90%) and test sets (10–15%). A sliding-window procedure consolidated each file into segments of three consecutive hourly

observations. For each segment, the Block Decomposition Method (BDM) calculated algorithmic complexity of the resulting 3×3 matrices while also computing the parameter differences across the three hours [317]. Each sample was represented by a 14-dimensional vector, forming the input for ML models. Because each file typically contained at least seven hourly observations per parameter, the final datasets are larger than original file counts.

ML models were trained using 10-fold cross-validation using the H2O platform [288]. GBM and stacked ensembles performed best; GBM was chosen for further analysis due to better interpretability. Figure 6.27 compares AUCs of three GBM models trained on data imputed by three methods:

- (a) LOCF/NOCB,
- (b) custom interpolation,
- (c) Kalman filtering combined with machine learning

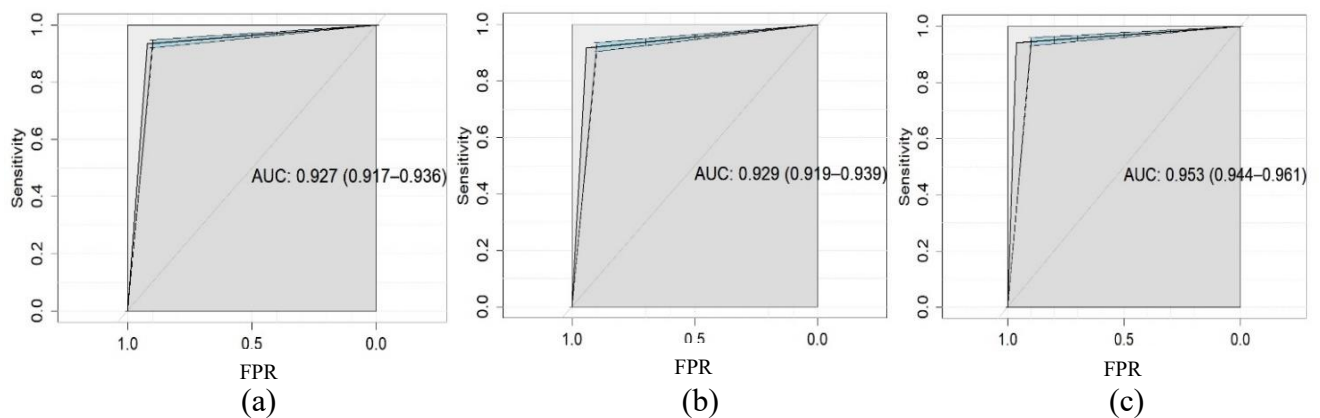


Figure 6.27 Classification performance, measured by the Area Under the Curve (AUC), of three GBM models constructed using data imputed through three different approaches: (a) LOCF/NOCB, (b) a custom method, and (c) Kalman filtering and machine learning. [278]

Model (c)—Kalman filtering for columns with partial missingness plus maximum-likelihood-based GLMs for entirely missing columns—achieved the best performance with an AUC of 0.953 on the held-out test set.

The Kalman-ML pipeline has two principal components: an adapted use of the Kalman filter to impute partially missing time-series columns and six GLMs used to recover columns that were entirely missing. This combination, applied to the current ICU sepsis dataset, produced the best predictive performance among tested methods. However, limitations exist: the approach has not yet been validated across many datasets (including those with categorical features or non-sepsis conditions), ignores uncorrelated variables, and does not explicitly account for differing variable distributions (normal, log-normal, etc.). Thus, datasets with continuous variables and similar inter-

feature correlations are most likely to benefit from this method. Future work should assess robustness across diverse datasets.

Table 6.17 provides detailed confusion matrices and performance metrics for the top GBMs.

Table 6.17. The confusion matrices and performance metrics for three distinct data recovery techniques [278]

		Reference (LOCF-NOCB)		Reference (Custom)		Reference (Kalman)	
		0	1	0	1	0	1
Prediction	0	1830	75	1874	89	1917	52
	1	159	1054	115	974	72	840
Accuracy		0.925		0.933		0.957	
95% CI		0.915 - 0.934		0.924 - 0.942		0.949 – 0.964	
P-Value		< 0.001		< 0.001		< 0.001	
Cohen's Kappa		0.840		0.854		0.900	
McNemar's Test P-Value		5.767e-08		0.080		0.088	
Sensitivity		0.920		0.942		0.964	
Specificity		0.934		0.916		0.942	

Note: The data in the table are reproduced from the author's work [278].

Figure 6.28 displays pairwise statistical comparisons (AUC) among the three approaches: (a) custom vs. LOCF, (b) Kalman vs. LOCF, and (c) Kalman vs. custom. The Kalman-ML approach significantly outperformed the other methods (AUC 0.953) with p-values of 7.4137e-05 (vs. LOCF) and 4.3727e-04 (vs. custom) [278].

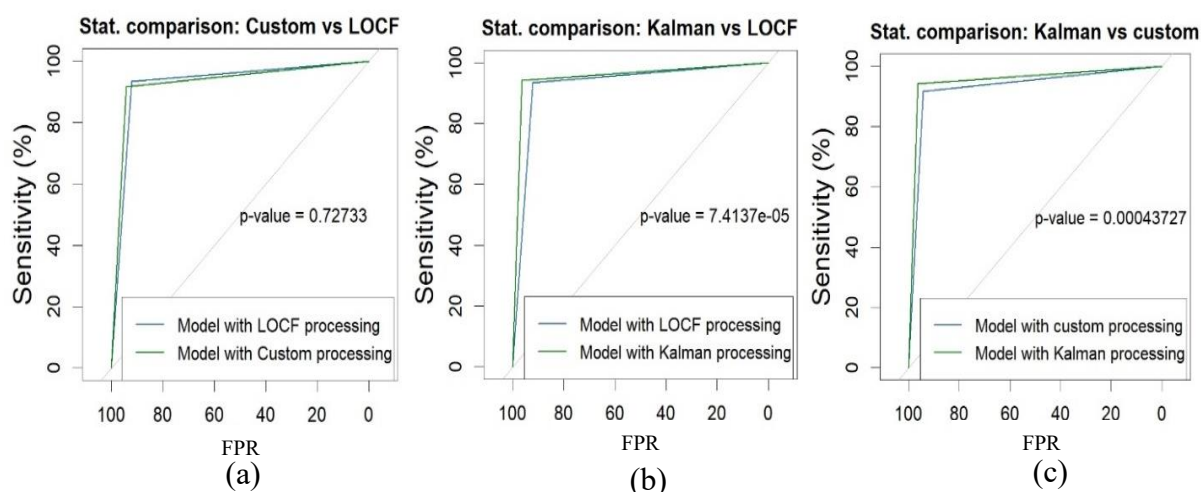


Figure 6.28. Statistical evaluation of the efficacy of three machine learning models, quantified by the area under the curve (AUC): (a) the custom approach compared to the LOCF method; (b) the Kalman method in relation to the LOCF method; and (c) the Kalman method versus the custom technique. Adapted from [278].

Further research directions include applying and comparing these imputation strategies on larger, analogous datasets to evaluate their generalizability and impact on sepsis vs. non-sepsis

discrimination. In the search for optimal imputation strategies for columns with a high fraction of missing values, several nonlinear regressions were tested (exponential, polynomial of various degrees) in addition to linear methods. Quadratic regression provided the best fit for these data; exponential regression performed less well (see Appendix 2). Given these findings and computational considerations, GLMs were selected for downstream analysis.

In summary, missing data are a significant challenge in large clinical datasets. Although many imputation strategies exist, there is no universal best method for ML applications. This work examines three distinct approaches, finding that the combined Kalman-filtering plus ML imputation yields superior sepsis-prediction performance (AUC 0.953) for the dataset studied. Because imputation choice can materially affect model discriminative ability, testing multiple methods to identify the most appropriate one is recommended. Improved imputation and representation methods can enhance diagnostic accuracy and patient care, potentially reducing mortality in conditions such as sepsis.

Additional machine-learning details will be provided in subsequent sections.

6.5. Synthesis of Chapter 6

This chapter documented preprocessing steps for the doctoral thesis, covering data preparation, algorithmic-complexity estimation, and representation-learning approaches to enable intelligent decision-support methods in intensive care settings.

Datasets

- EEG Epileptic Set: Intracranial EEG recordings from epileptic dogs and humans, segmented into 10-minute multichannel clips labeled preictal or interictal (e.g., Dog-2: 500 interictal and 42 preictal samples, 16 electrodes, 400 Hz).

- EEG Non-Epileptic/Healthy Set: Recordings from 36 healthy subjects performing mental arithmetic, with 20 EEG channels and ECG at 500 Hz (Figure 6.1 shows 10-channel EEG examples).

- Sepsis Set: Hourly time series of 40 parameters (vital signs, labs) from >40,000 ICU patients. Set A (20,336 cases, 1,790 sepsis) was selected for fewer missing values (Figure 6.2; Table 6.1).

Missing data handling

- LOCF/NOCB for partial gaps.
- Custom interpolation (e.g., R zoo) for partial gaps and Gaussian-based imputation for fully missing columns.

- Kalman filtering for columns with ≥ 3 non-missing values combined with GLMs for fully missing columns (Tables 6.2–6.5; correlation analysis in Table 6.3 and regression coefficients in Table 6.4). Figure 6.3 diagrams the data-processing flow. Data were filtered to ensure at least seven hourly observations per file for a four-hour prediction horizon.

Algorithmic complexity estimation

- Kolmogorov–Chaitin complexity approximated via Block Decomposition Method (BDM), part of Algorithmic Information Dynamics.

- For EEG: multichannel signals were formed into matrices (e.g., 16×16), binarized (row-mean thresholding), and BDM computed per matrix to produce complexity time series (Tables 6.6–6.8; Figure 2.4).

- Text conversion: numeric BDM values were tokenized into text and analyzed with LDA topic models (Figures 6.5–6.7), revealing topic overlap differences between interictal and preictal segments.

- For sepsis: 3×3 matrices per sliding window were BDM-scored (Tables 6.9–6.11), and final ML-ready data tables include 14 features (BDM values and temporal differences) as in Table 6.12.

Representation learning and NLP tools

- Textual methods: TF–IDF for importance weighting; embeddings: Word2Vec (CBOW/Skip-gram), fastText (subword n-grams), BERT (short-sequence models), and BiLSTM (Figure 6.10 and Figure 6.17).

- Data splits for Word2Vec/fastText/BiLSTM are in Table 6.13; BERT splits in Table 6.14.

- Embedding diagnostics: UMAP colored by PCA-based global structure (Figure 6.15); PCA on AC tokens grouped words by complexity (Figure 6.16).

- Nonlinear embeddings (LLE, MDS, Isomap, t-SNE, spectral embedding) revealed improved separability in arithmetic-test visualizations (Figures 6.11–6.14) and produced various projection views for sepsis (Figures 6.24–6.26).

Intermediate evaluation

- Imputation methods materially changed sepsis-prediction performance. GBM models trained on LOCF/NOCB, custom, and Kalman-ML-imputed data achieved AUCs of 0.925, 0.933, and 0.953, respectively (Figure 6.27). Table 6.17 and Figure 6.28 provide confusion matrices, performance metrics, and statistical comparisons. Kalman-ML produced the best accuracy (0.957), sensitivity (0.964), and specificity (0.942).

Conclusions

- Preprocessing steps—BDM-based complexity estimation, text conversion, embeddings, and considered imputation—improved data separability for epilepsy classification and sepsis prediction.
- The Kalman-filtering plus ML imputation approach performed best on the evaluated ICU dataset but requires further validation across diverse datasets and feature types.
- Overall, combining algorithmic-complexity measures with representation-learning techniques provides a promising foundation for complexity-aware intelligent systems in intensive care, with the potential to improve diagnostic accuracy and patient outcomes.

7. INTEGRATING ALGORITHMIC COMPLEXITY IN ML FOR INTENSIVE CARE APPLICATIONS: FINDINGS, DISCUSSIONS AND PRACTICAL IMPLICATIONS

7.1. Stages of machine learning system development. Techniques and algorithms used

7.1.1. Main elements of a machine learning cycle

The general procedure to build a machine learning model is well defined (see Figure 7.1). .

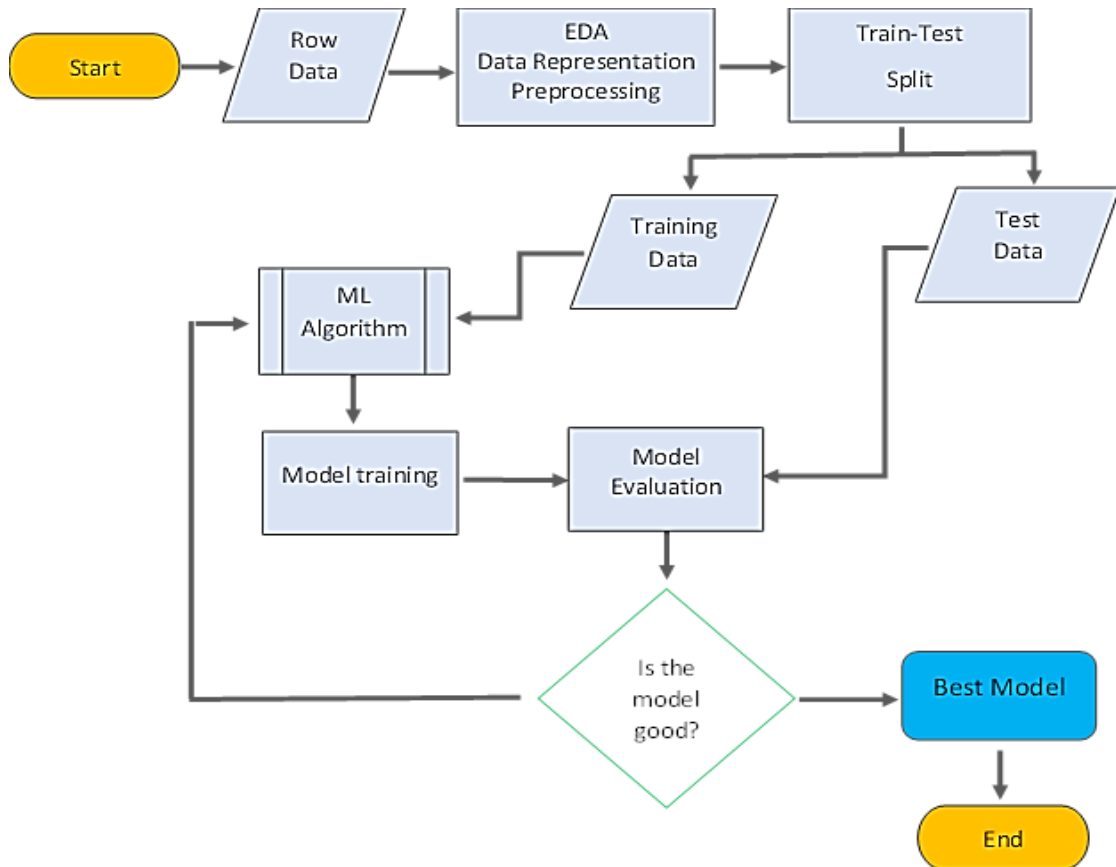


Figure 7.1. Main elements and steps in the creation of a machine learning system.

After preliminary data processing a subset is first separated for testing (here we call it sepsis test set) which contains 2881 samples (892 sepsis cases and 1989 non-sepsis cases), excluded from training and validation sets (Figure 7.2). EEG set goes through the same steps.

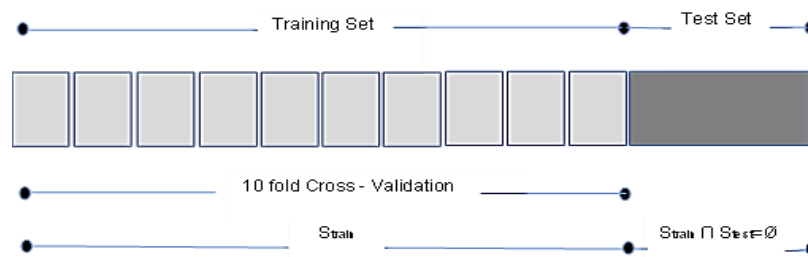


Figure 7.2. Partitioning of data into training and test sets.

Note: The data is first split into two sets (training set used for learning and cross-validation and test set).

7.1.2. Governing aspects of ML for medical purposes

Many critics say that research which shows how prediction models are developed and validated fails to provide information which readers need in order to judge the methodology and understand what the model predicts accurately and other important things about it. TRIPOD guideline was established in 2015 to address this problem (poor and partial reporting of multivariable prediction models for individual prognosis or diagnosis). Without good reporting no research outcome can be implemented (validation studies, evidence synthesis studies or routine practice) [320].

Since TRIPOD came into being publications in these disciplines increased dramatically. Artificial intelligence (AI) especially machine learning has developed and become popular in clinical forecasting methods. Data on quality of reporting of this research cannot be obtained because recently machine learning methods have been applied to clinical prediction modeling. However, only a few of these assessments that looked at the quality of reporting found it to be poor [321]. Protocols need to be developed for writers who use machine learning methods to give a full account of their predictive model research [320].

TRIPOD group have launched a global initiative called TRIPOD-AI [320] which aims to develop consensus driven extension of TRIPOD where documentation of research relating to the development (validation or improvement) of diagnostic or prognostic prediction models using machine learning is emphasized. TRIPOD-AI extension should guide researchers, authors, publishers, users and other stakeholders on what basic elements to include in reports; checklist and document with explanations and examples of good reporting for every article. Good reporting is expected to increase understanding, replication and use of research.

In 2019, the Prediction Model Risk of Bias Assessment Tool (PROBAST) was developed, spanning multiple domains, to provide a detailed analysis of the approaches. It also has 20 probing questions to help assess bias risk. Assessment and documentation of bias risk are closely related

since assessment of bias risk depends on the results of main research [321]. Therefore, these studies must be critically evaluated before adoption even though PROBAST can be used for predictive model investigations that use machine learning (although there has been progress in evolving and validating model building techniques — nomenclature included). TRIPOD-AI should be developed alongside PROBAST-AI so that they complement each other. The former is about research or projects that involve creation, improvement or verification of a multivariable prediction model using different machine learning methods while the latter deals with bias assessment in research involving multivariable prediction models (using machine learning).

Both TRIPOD-AI and PROBAST-AI address research or projects that focus on creating, improving or verifying a multivariable predictive model using various machine learning methods. In the first TRIPOD and PROBAST documents a multivariable prediction model is defined as a formula or combination of several predictors used to predict whether a person will experience a certain health outcome or condition (e.g., diagnosis or prognosis). The patient's medical history, physical exam, diagnostic work-up, prognostic study or monitoring test (and treatments) may all be indications. Results can also take dichotomous, categorical or continuous forms and include an extensive range of possibilities (e.g., identification of a particular disease or disorder (diagnostic or classification result), immediate prognostic elements such as in-hospital mortality or postoperative problems, long term prognostic elements such as treatment complications after one year, metastases after five years or lifetime survival rates).

The distribution of these study results is important too. The original goal of the “Standards for Reporting Diagnostic Accuracy” (STARD) statement was to improve reporting quality in diagnostic accuracy research. In 2015 a revised set of 30 critical elements that must be included in every diagnostic accuracy study report was published. This revision aims at increasing thoroughness and lucidity of reporting in diagnostic accuracy studies by integrating new findings on what might cause bias and variability in diagnostic precision [322]. It's about making STARD work better. STARD 2015 did not take into account the difficulties and complexities of AI interventions. In 2021 a new tailored version of STARD checklist (called STARD-AI) was introduced to improve reporting on research on accuracy of diagnostic tests using AI or AI technologies. The present research also takes into account suggestions made in these directives.

7.1.3. Criteria used to evaluate the performance of machine learning model

Because metrics are the most important ones to evaluate the performance of the completed products and systems developed (and since in this research we combine metrics from artificial intelligence and statistics, they are explained below [324, 325]:

$$\text{Sensitivity } \left(\frac{\text{Recall}}{\text{TPR}} \right) = \frac{TP}{(TP+FN)}, \quad (7.1)$$

where, TPR – true positive rate, TP – true positive (results), FN is false negative (results);

$$\text{Specificity (True Negative Rate)} = \frac{TN}{(TN+FP)}, \quad (7.2)$$

where, TNR – true negative rate, TN – true negative (results), FP is false positive (results);

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)}, \quad (7.3)$$

where, TP – true positive (results), TN – true negative (results), FN – false negative (results), FP – false positive (results);

$$\text{Precision (Positive Predictive Rate)} = \frac{TP}{(TP+FP)}, \quad (7.4)$$

where, TP – true positive (results), FP – false positive (results);

$$\text{Negative Predictive Rate} = \frac{TN}{(TN + FN)}, \quad (7.5)$$

where, TN – true negative (results), FN – false negative (results);

$$\text{False Positive Rate} = \frac{FP}{(FP+TN)}, \quad (7.6)$$

where, FP – false positive (results), TN – true negative (results);

$$\text{False Negative Rate} = \frac{FN}{(FN+TP)}, \quad (7.7)$$

where, FN – false negative (results), TP – true positive (results);

$$\text{Positive likelihood ratio (LR +)} = \frac{TPR}{FPR}, \quad (7.8)$$

where, TPR – true positive rate, FPR – false positive rate;

$$\text{Negative likelihood ratio (LR -)} = \frac{FNR}{TNR}, \quad (7.9)$$

where, FNR – false negative rate, TNR – true negative rate;

ROC analysis (*ROC* - Receiver operating characteristic - as ratio between FPR (false positive rate) on X axis and TPR (true positive rate) on Y axis, where $FPR = \frac{FP}{(FP+TN)}$, and $TPR = \frac{TP}{(TP+FN)}$, including the area under the *ROC* curve *PR analysis* (*PR*—Precision-recall – as ratio between sensitivity/recall on X axis and positive predictive value/precision on Y axis, a method better suited for unbalanced data); *Confusion matrix* (information about "actual" and "predicted" cases); *Variable importance analysis* (most important predictors); *Loss function analysis* (during model training).

Metrics related to the Kolmogorov-Chaitin complexity have a special role and come from the field of algorithmic information dynamics, as described in Chapter 6.

7.2. Training, testing, validation and comparison of developed systems

7.2.1. Algorithmic complexity of the EEG time series (numeric representation) for exploratory data investigation

The algorithmic complexity (AC) time series obtained from EEG signal data can provide interesting information about the data but for long EEG segments this is not useful because there are too many points in the figure (e.g., Figure 6 .11).

One can aggregate the AC values by taking the mean of the time series over non-concurrent windows of a given width (e.g., Figure 7.3 shows the amalgamated AC time series for an “interictal” segment (blue) and a “preictal” segment (red)).

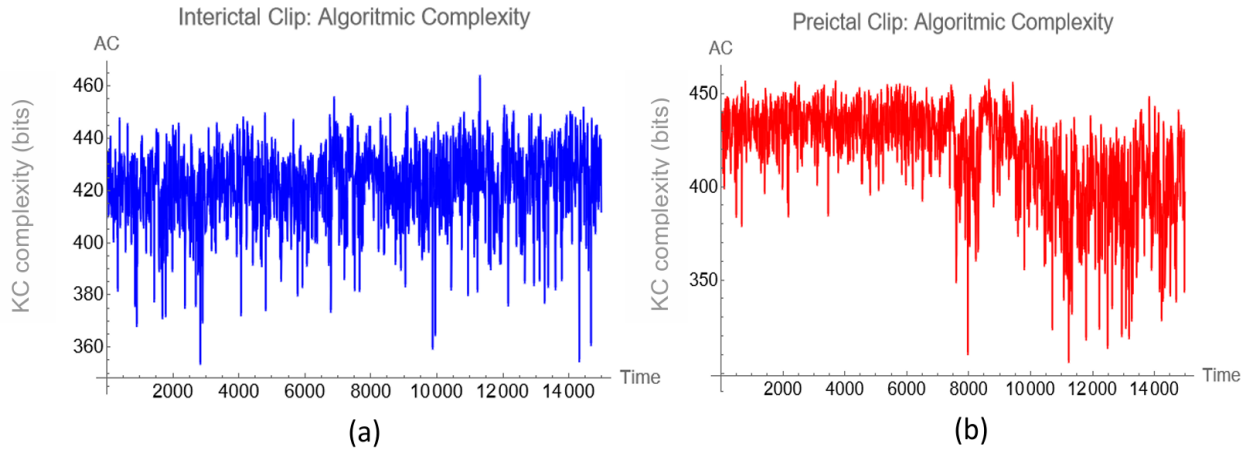


Figure 7.3. Algorithmic complexity (AC) over time : (a) for a “normal” /interictal EEG segment and (b) for a “pre-epileptic”/preictal segment.

The two diagrams are not similar when compared together: the first part of the preictal clip has higher AC values (around 430) but also less variation.

One can draw a picture that shows exactly how different the AC value traces of EEG segments from two groups (interictal vs preictal) are. But this is only true under normal conditions; most segments cannot be classified directly by such a picture. Exploratory data analysis by itself is limited by the amount of information it provides (see above); in this case machine learning can be helpful (as we will see later).

7.2.2. Using modified EEG data for machine learning

After adaptation to ML algorithms, EEG data is given in two forms: (a) TF-IDF data and (b) NLP models data. Tables 7.1 and 7.2 show these formats.

Table 7.1. Data for machine learning using TF-IDF

Cat	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	...	V1100
0	0.0	4.0	1.4	0.0	0.2	0.3	1.5	0.0	0.0	3.3	-2.7	...	0.0
0	0.0	1.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0

0	0.0	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
1	0.0	5.3	0.6	-1.2	0.2	0.0	0.0	0.0	1.9	0.0	4.0	...	0.0
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0

Note: “Cat” column denotes the sample label: “0” – interictal/normal EEG clip and “1” – preictal clip. V1 to V100 are the beta-log ratio of TF-IDF.

Table 7.2. Information for machine learning from NLP models

Cat	Text
1	chi dzi czd dcd dbz cbh cid caz dba dcd dgd dbc dce dzi dbd dae dbc dca dei ...
0	afc ahg aiz bii bbi ahi adz bai gh ee cea cez dde cii che cee bff bfb cfb ahi dzd ...
1	dai def ceb dde ciz cec dec dzc dfe ddh def dad dbh dbh dbd dee daf dea dda ...
0	dce chh cid dbc dbe dea dbf cha dfd deg dch dab dag chh dfa ddb dei dae dda ...
0	ddi dei ddb ceg ded ddh dag dbz dfe dbg dga dbg ddf ddg ddg dga dez def dde ...

Note: “Cat” column is the sample label (“0” – interictal/normal EEG clip; “1” – preictal clip). Words are KCC values converted to letters/words.

Term Frequency-Inverse Document Frequency (TF-IDF)

The formatted preprocessed data (Table 7.1) were then fed into a model based on GBT (Gradient Boosted Trees). This program used decision trees and made predictions (classification between normal/interictal vs. preictal segments). Since the dataset was very unbalanced in this machine learning study we generated additional data by randomly assigning a number in [-1, 1] to each element of the original AC values.

In the initial stage of machine learning the original dataset was split into two subsets: (a) training subset (440 interictal clips and 37 preictal clips) and (b) testing subset (59 interictal clips and 5 preictal clips).

The new data-generation method was applied only to the training set and so the number of preictal clips (in AC format) increased from 37 to 443. The final training set for machine learning consists of 440 "interictal" subsets and 443 "preictal" subsets, while in the test set there are still 59 "interictal" subsets and 5 "preictal" subsets. Figure 7.4 shows how good is our established system on Wolfram Language 12.3 platform. In Figure 7.4 the confusion matrix is shown with a standard threshold value of 0.5. Classification of preictal clips is obviously correct but interictal clips classification is not so good.

If we change the classification threshold to 0.65 then all preictal clips are correctly classified as preictal and misclassification of interictal clips reduces from 22 to 5 which gives a conclusive diagnostic precision of 0.92 .

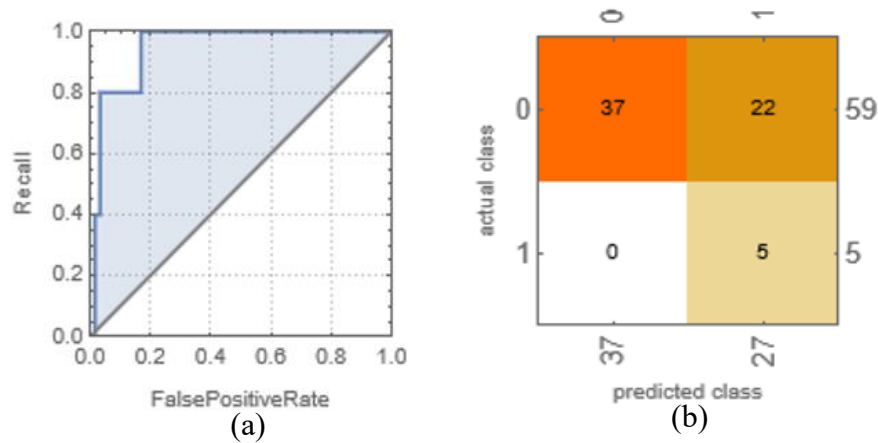


Figure 7.4. The ROC curve (a) and the contingency table (b) of the trained model.

Word2Vec

This model was implemented on H2O.ai with parameters given in Table 7.3.

Table 7.3. Variables utilized in the development of the Word2Vec model [343]

Variable/ Parameter	Specification of the parameter	Value of the parameter
Dimensions	Count of vector dimensions	100
Window	Contextual lexeme magnitude	5
Workers	Count of concurrent threads	4
Min. count	Minimum word count	1
Model	Word2Vec model type	CBOW/Skip-Gram

Note: CBOW – continuous bag of words

The results of using a skip-gram variant classifier based on Word2Vec on Dog 1 and Dog 2 are shown in Figures 7.5 and 7.6 respectively.

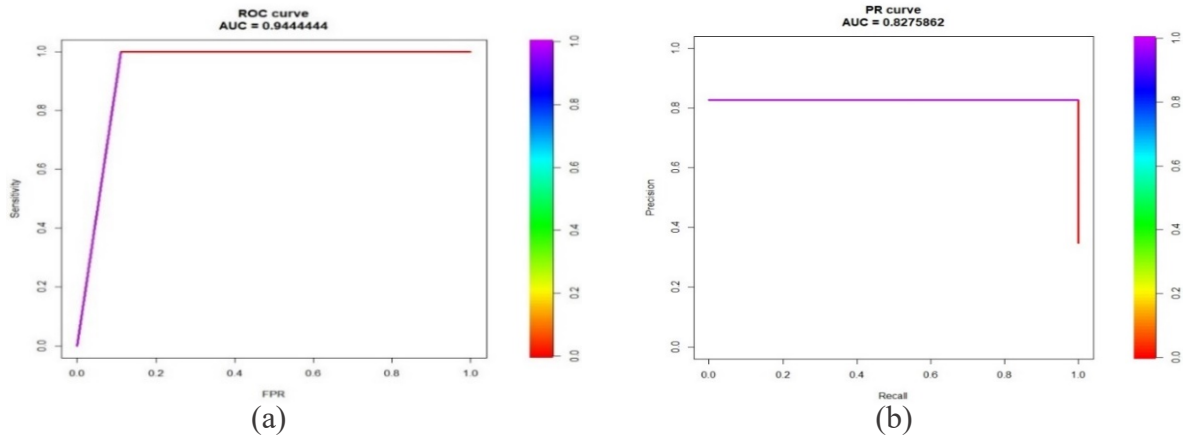


Figure 7.5. The efficacy measured by AUC (a) and Precision-Recall AUC (b) of Skip Gram Word2vec model on Dog 1 test.

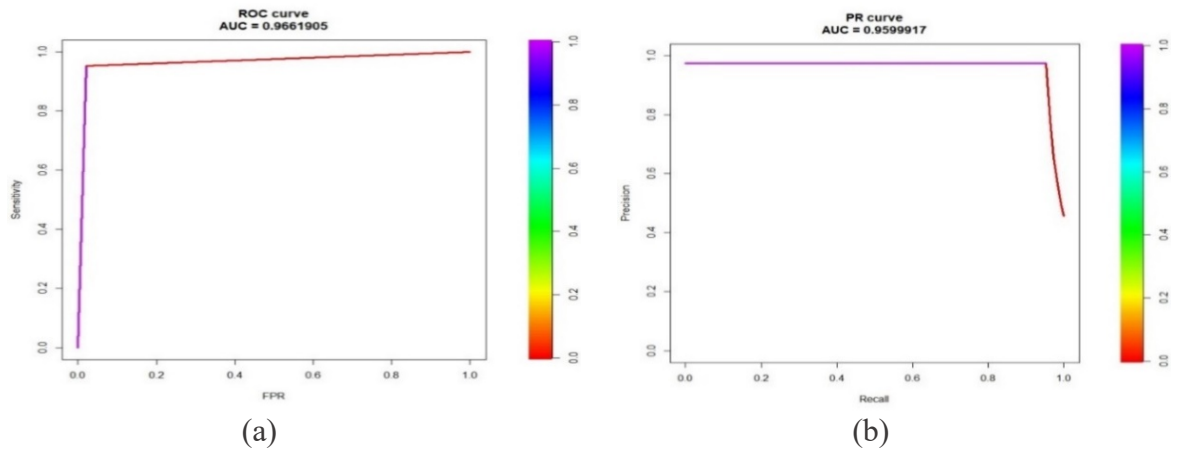


Figure 7.6. The efficacy measured by AUC (a) and Precision-Recall AUC (b) of a Skip-Gram Word2vec model on the Dog 2 test dataset.

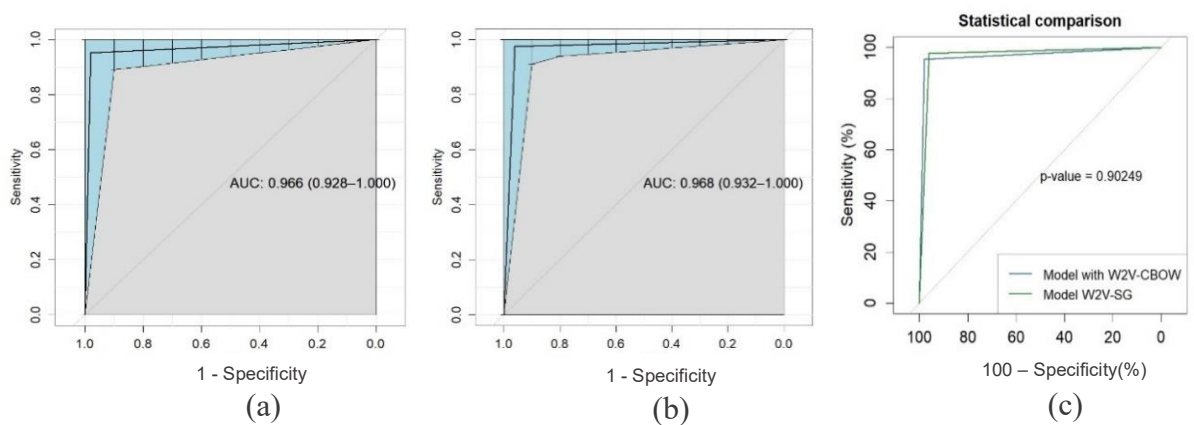


Figure 7.7. Comparison (c) between CBOw (a) and Skip-Gram (b) models.

Both versions of word2vec CBOw and Skip-Gram performed similarly on the Dog 2 data set (Figure 7.7) with no statistically significant difference ($p = 0.90249$).

fastText

The fastText model has intrinsic capability to do supervised machine learning on text. This is available in both Facebook and Gensim libraries. A model was built from scratch with an EEG dataset using FB fastText program. Figure 7.8 shows the performance of top models when different hyperparameters were tuned (learning rate, vector dimensions, epochs, wordNgrams (“on” and “of”) and autotuning).

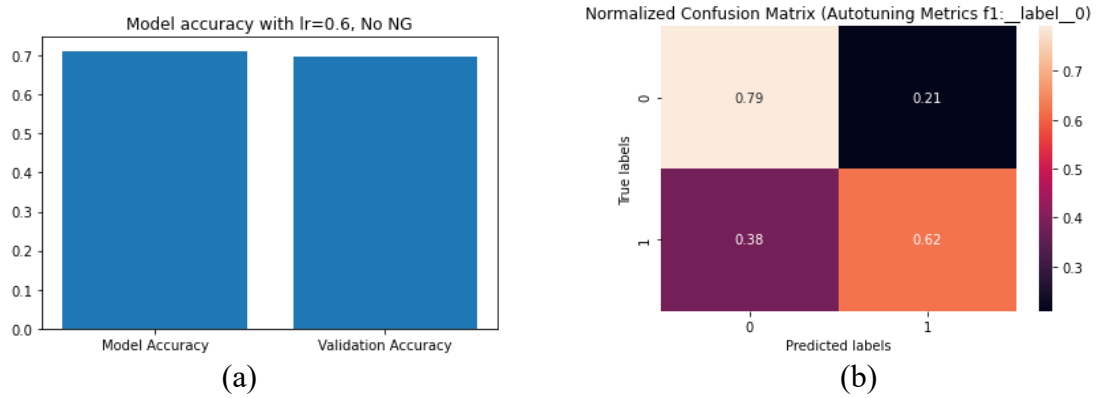


Figure 7.8. The performance of a fastText model on EEG data set: (a) – model accuracy, (b) – the contingency table (confusion matrix).

The wordNgrams function was omitted because we expected to get optimal performance without it. Performance also does not vary much if you compare 20 vs 50 vectors. They are so unique that it was impractical to deactivate lemmatization and stemming.

BERT

In the research involving the BERT model a sentence usually has around 500 words in it. A few short sentences were excluded. Development and training of a BERT model is similar to word2vec model: (a) early training stage on unannotated data and (b) later fine-tuning stage when model is refined on annotated data. The design of the BERT model used in our study and the BERT classifier are given in Annex 3. The settings for the most efficient BERT model we tested were:

```
attention dropout rate: 0.1,  
hidden activation function: "gelu",  
hidden dropout rate: 0.1,  
hidden size: 768,  
initializer range: 0.02,  
intermediate size: 3072,  
layer normalization coefficient (eps): 1e-05,  
maximal position embeddings: 514,  
number of attention heads": 12,  
number of hidden layers": 6
```

The only difference between models built for the dog 1 and dog 2 sets is the number of “distinct words” in vocabulary (734 words in dog 1 set vs. 803 words in dog 2 set). Figures 7.9 and 7.10 show the performance of the model.

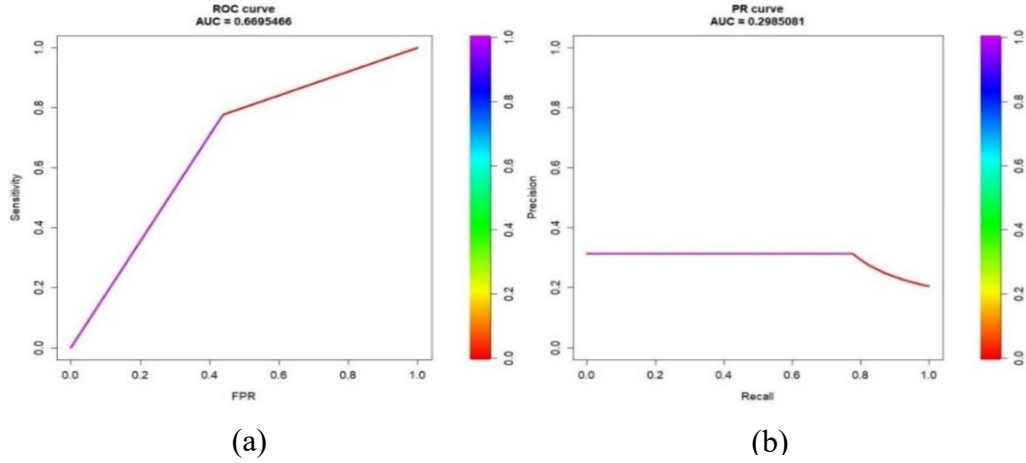


Figure 7.9. The efficacy measured by AUC (a) and Precision-Recall AUC (b) of a BERT model on the Dog 2 test dataset.

Unlike large sequences (for example, of 15,000 words in an EEG clip), BERT uses shorter peaces of roughly 500 words (or AC values translated to words). This means that classification of these short sequences might be wrong. To verify this and correct it we can first classify the short sequences using a Word2vec classifier; then after denoising these newly labeled sequences can be fed into a BERT based classifier. As shown in Figure 7.10 such a situation improves the classification performance by approximately 8% with area under ROC curve.

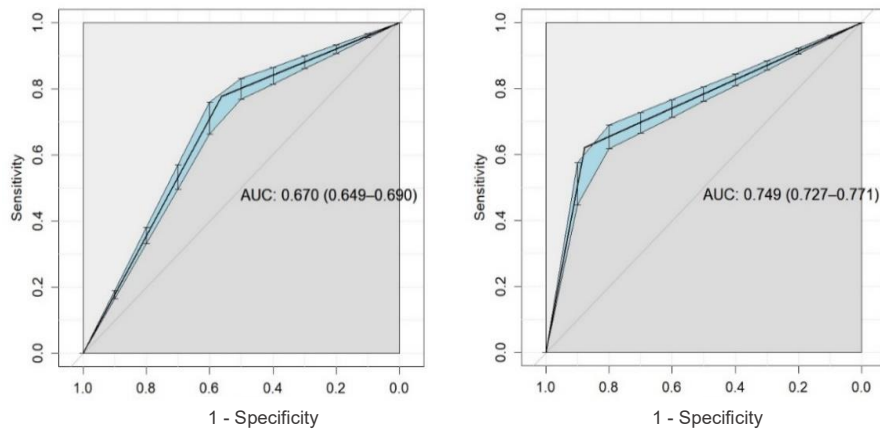


Figure 7.10. Performance (by AUC) of a BERT model on the Dog 2 test set: left - “pure BERT”; right - BERT where train and test data were “denoised” by w2v models before BERT training/fine-tuning and testing.

Visualization techniques can help to understand the internal dynamics of the model. The Python BertViz package [326] is used to produce Figures 7.11 and 7.12 showing various general and specific features of the model built on Dog 2 data. Appendix 3 lists the study’s BERT model architecture and classifier.

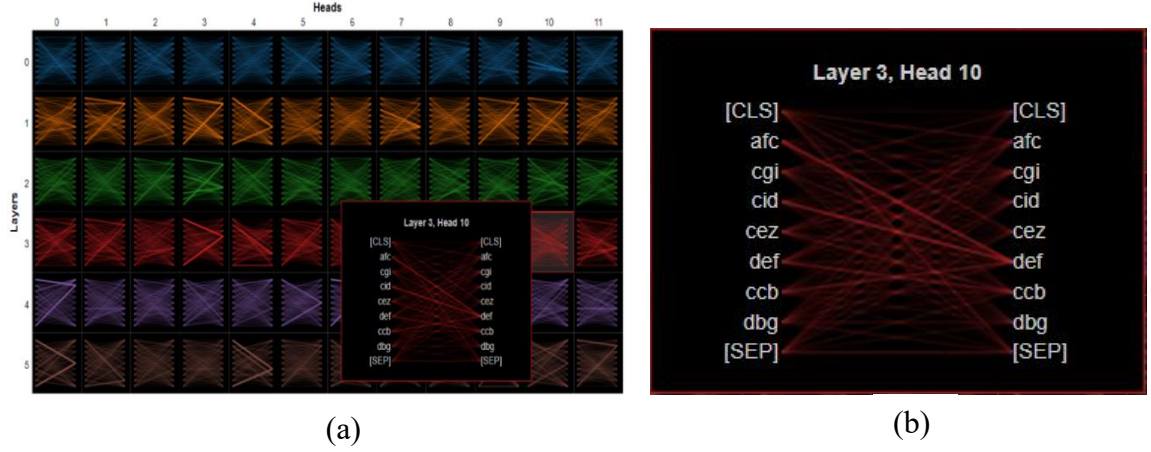


Figure 7.11. Visualizations of the BERT model built on the Dog 2 test set. On the left (a) there is a general view of layers and heads. Image on the right (b) shows relationships between words as an attention map.

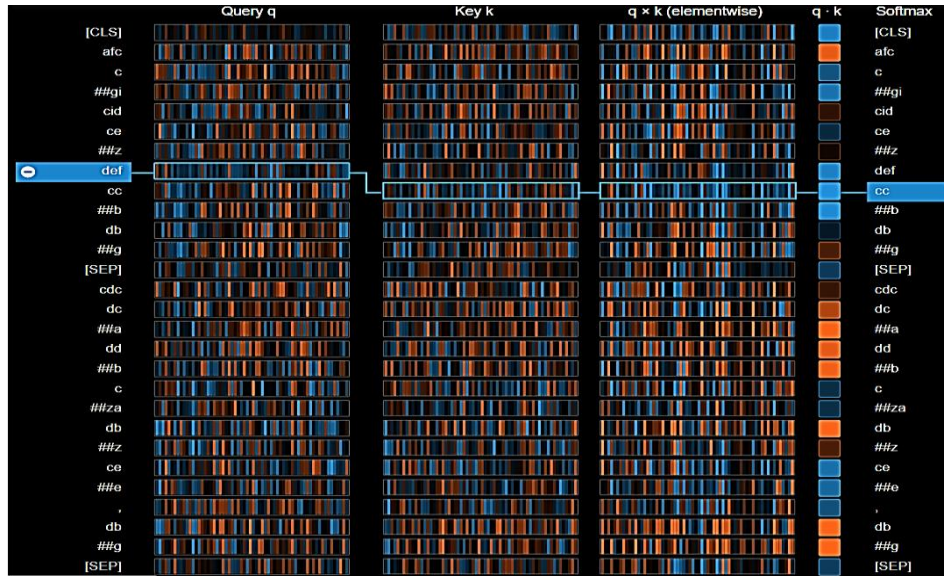


Figure 7.12. Visualization of a particular layer (#2) and head (#0) for a particular pair of tokens (def → cc) as heatmaps.

BiLSTM

The PyTorch version of the model has the following architecture:

```
BiLSTM_eegAnalysis(
    (embedding): Embedding(468, 64, padding_idx=0)
    (lstm): LSTM(64, 32, batch_first=True)
```

```
(dropout): Dropout(p=0.2, inplace=False)
(fc): Linear(in_features=32, out_features=3, bias=True)
)
```

We trained the model for 300 epochs and reached stability after 110 epochs (Figure 7.13).

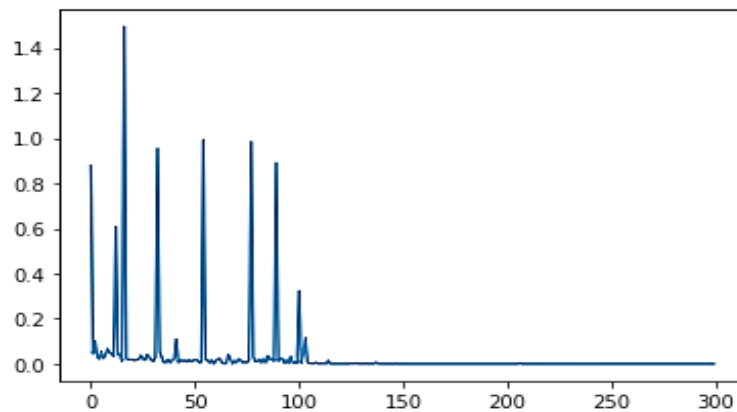


Figure 7.13. Loss of the PyTorch BiLSTM during training with Dog 2 data. On X axis are represented the epochs and on Y axis is loss.

Accuracy of this model on the test set of Dog 2 is 0.891.

Figure 7.14 shows the performance of a TensorFlow Keras model with an identical architecture to the PyTorch model above (12 epochs training). Increasing the number of epochs did not change accuracy — so increasing the number of epochs did not help. Although Keras model is accurate in training (more than 0.980), its test set performance (green line) is much worse than that of the PyTorch model.

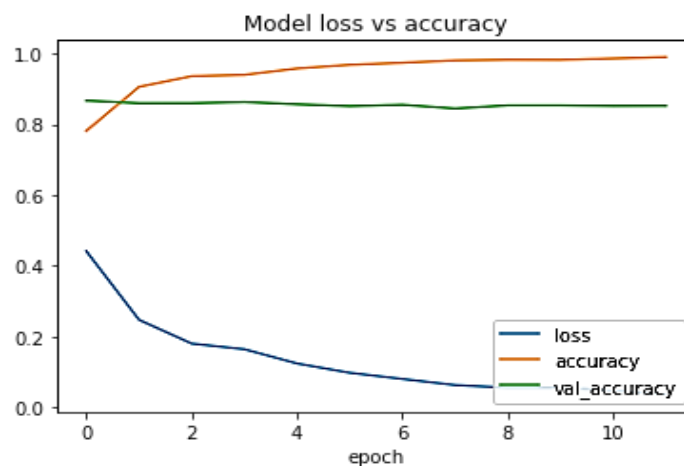


Figure 7.14. Training and validation results of the Keras model with Dog 2 data.

Table 7.4 summarizes the most successful models for dog 2 test set in terms of area under the receiver operating characteristic (ROC) curve or diagnostic accuracy.

Table 7.4. The most effective NLP models built with dog 2 EEG data

NLP model	Evaluation of model performance on test set
Word2Vec (CBOW)	AUC 0.97
Word2Vec (Skip-Gram)	AUC 0.97
BERT	AUC 0.67
BiLSTM	Accuracy 0.89
TF-IDF	Accuracy 0.92

Note: CBOW – continuous bag of words, AUC – area under the ROC curve

For Dog 1 test set the efficacy of the models was similar. The best performing model was Word2Vec with an area under ROC curve of 0.944 (Figure 7.5).

Our studies present an idea for biological signal processing and medical applications (EEG clip categorization). Data analyzed with algorithmic complexity (AC) is not currently shown over time. Algorithms are a manifestation of the mental activity going on in our brain. The promising results presented here seem to indicate that we should not be overly optimistic until the system has been fully implemented in clinical settings and unforeseen technical problems accounted for. We should also note that it is unwise to make too many predictions before the system has been fully tested in clinical settings and any unexpected technical problems have been considered.

The Kolmogorov-Chaitin algorithmic complexity structure is a significant mathematical framework for analyzing EEG data as it allows us to identify unique features in brain activity patterns. EEG analysis can provide great insight for research and therapeutic applications because we can identify unique brain activity patterns associated with cognitive functions or neurological disorders.

To achieve this goal more research is needed to automate extraction of standardized clinical information from EHRs and patient interviews (especially in the case of epilepsy). NLP is commonly used in epilepsy studies to recognize patients, classify risk levels and predict results. This approach uses traditional natural language processing methods for textual data. We study a new approach: using NLP algorithms/models on modified EEG data. The textual final format of data makes it amenable to NLP technologies. Among NLP models used in this way the Word2Vec model performed best.

The CBOW version of this algorithm had an impressive AUC of 0.968. This model is the one we will be looking at more in detail since it was used to predict epileptic seizures. Although the results of this research are promising, there is still a lot of work to do in order to build a reliable seizure forecasting system. This will be explored in further research works.

The reason Word2Vec model performs better than BERT is that it can handle long texts (a full 10 minute EEG recording transcribed into text). Long texts may have important connections between words which are far apart but which are relevant for classification.

7.2.3. Using the sepsis dataset for machine learning applications

In section 6.2.3 we explained how our method solves the categorization and forecasting problem in sepsis by converting time series data into vectors: they consist of two elements (a) algorithmic complexity (as measured by BDM) on a matrix of physiological systems/processes (b) temporal dynamics of relevant predictor values (V3 to V14), that is variation in these parameters between samples. Table 6.12 shows five vectors (instances).

Machine learning approaches to sepsis data [316]

The dataset (Table 6.12) was structured as above but had a column with labels 0 for non-septic and 1 for sepsis. These were passed to (a) GLM, (b) GBM, (c) DRF and (d) DNN algorithms which performed a binomial classification task (sepsis vs non-sepsis).

The optimal hyperparameters were found by AutoML (H2O preset platform): number of predictors for GLM, learning rate, number of trees, depth for GBM and DRF; number of branches; layers and activation functions for DNN as multilayer perceptron. The hyperparameters were then checked in the current dataset using a 10-fold cross-validation grid search. Optimal configuration of major hyperparameters for one algorithm/model is given in Table 7.5.

Table 7.5. Best main hyperparameters: 4 ML algorithms/models

Algorithm/model	Optimal configuration for major hyperparameters
GLM	“Binomial” type with 14 predictors, 34 iterations
GBM	252 trees, 6 – mean depth, 55.52 - average number of leaves
DRF	50 trees, 20 – mean depth, 3729.68 - - average number of leaves
MLP	4 layers, 2 hidden with 50 units each, Rectifier with Dropout (20/10/10) – activation; Softmax—for output

Note: GLM - generalized linear model, GBM - gradient boosting machine, DRF - distributed random forest, MLP - multilayer perceptron.

In this research the procedure to build machine learning models followed a conventional protocol: identification and collection of data, preprocessing (if necessary), algorithm selection according to dataset type; model training, validation and testing; performance evaluation. An important step is the preliminary formation of test set (10% of total data) and 90% for training and validation.

After exploratory analysis of the sepsis dataset, we collected a total of 28,995 samples. These were then divided into training set (26,114 samples) and test set (2,881 samples). Selection criteria were (a) at least 7 hourly observations (4 hours for prediction and 3 hours for fluctuations evaluation) and (b) high data density (for non-septic patients: 7 continuous measurements for each parameter of interest). The missing points for septic patients were recovered using the algorithm described in section 6.1.2.

We fed the same dataset to all four algorithms/models (GLM, GBM, DRF and DNN) with the hyperparameters given in Table 7.5. Figure 7.15 and Table 7.6 show how these methods work.

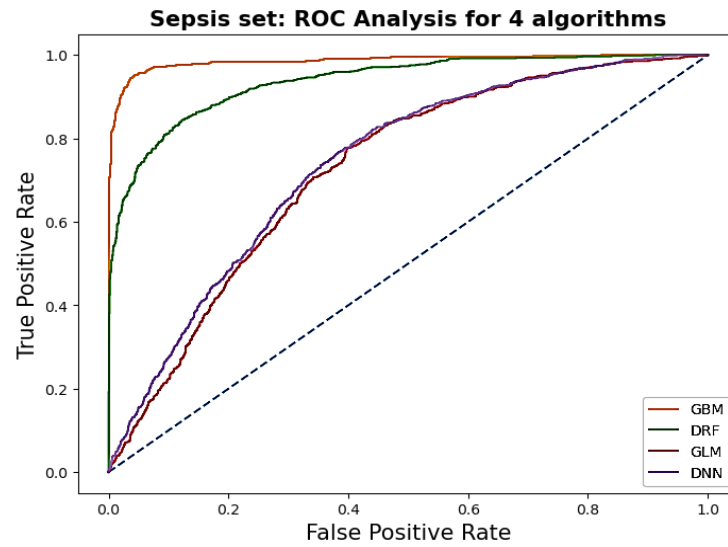


Figure 7.15. Receiver Operating Characteristic curves for four different algorithms (DNN, GBM, GLM, DRF).

In Figure 7.15 we can see that the most powerful model is GBM (gradient boosting machine) followed by DRF (distributed random forest), but GLM and DNN have almost equal performance. A more detailed analysis of the performance is given in Table 7.6.

Table 7.6. Performance of the 4 ML algorithms/models

Metrics/Model	GBM		DRF		GLM		DNN	
	Cross-validation (sd)	Test (95% CI)	Cross-validation (sd)	Test (95% CI)	Cross-validation (sd)	Test (95% CI)	Cross-validation (sd)	Test (95% CI)
Area under the ROC curve (AUC)	0.974 (0.003)	0.953 (0.944 – 0.961)	0.920 (0.009)	0.827 (0.812 – 0.843)	0.707 (0.010)	0.684 (0.667 – 0.700)	0.745 (0.048)	0.539 (0.530 – 0.547)
Recall	0.892 (0.010)	0.942 (0.924 – 0.956)	0.753 (0.021)	0.685 (0.653 – 0.715)	0.820 (0.035)	0.831 (0.805 – 0.855)	0.768 (0.064)	0.977 (0.964 – 0.985)
Specificity	0.971 (0.007)	0.964 (0.955 – 0.972)	0.917 (0.016)	0.970 (0.961 – 0.977)	0.528 (0.041)	0.537 (0.515 – 0.559)	0.619 (0.114)	0.101 (0.088 – 0.115)

Table 7.6. Continued

Accuracy	0.946 (0.004)	0.957 (0.949 – 0.964)	0.869 (0.009)	0.882 (0.869 – 0.893)	0.620 (0.019)	0.628 (0.610– 0.646)	0.666 (0.059)	0.372 (0.354– 0.390)
Precision (PPV)	0.934 (0.015)	0.921 (0.902 – 0.938)	0.809 (0.029)	0.911 (0.886 – 0.931)	0.443 (0.013)	0.446 (0.422– 0.470)	0.489 (0.062)	0.328 (0.310– 0.346)

Note: Performance is shown for cross-validation as well as test set. CI – confidence interval.

The sepsis prediction system developed in this study was based on systems identified in the literature as the best performing ones. Table 7.7 shows performance metrics of the top five winning systems from the 2019 Challenge which is used for our research (data source). It also lists performance metrics from International Guidelines for Management of Sepsis and Septic Shock 2021 and the performance of the system developed in current study.

Table 7.7. Comparison of several sepsis forecasting systems

Metrics	2019 Challenge (the first 5 results)	2021 Guidelines (95% CI)	Created system (95% CI)
AUC	0.83-0.87	0.89 (0.86 - 0.92)	0.95 (0.94 - 0.96)
Accuracy	0.79-0.90	-	0.96 (0.95 - 0.96)
Recall	0.65*	0.81 (0.80 - 0.81)	0.94 (0.92 - 0.96)
Specificity	0.88*	0.72 (0.72 - 0.72)	0.96 (0.96 - 0.97)

* performance of the winner model; CI – confidence interval.

It has been implemented into Shiny application (R programming language) and desktop version (Python) for clinical use [327].

The high performance of GBM sepsis prediction system makes it suitable for clinical use in Republic of Moldova. This device can help patients by alerting doctors about sepsis threat coming soon. With such an early alert antibiotic treatment can be started immediately which is studied and recorded in academic.

7.3. Explainability/Interpretability aspects of created models

How a user adopts a model depends on how much he knows about its results, how aware he is about them and how clear/comprehensible they are to him. In this case clarity and comprehensibility of the model are very important.

7.3.1. Models using EEG data

NN-based visualization. ML model explainability/interpretability

To increase transparency of architecture of large neural networks we need to provide complete explanation of input processing and output of every model. This is how each attribute

contributes to the final outcome. This step is necessary regardless of what field (NLP, speech processing or computer vision) it is in. Historically this contribution was explained using variable significance [328]. But because many new neural architectures are emerging quickly a richer representation became necessary.

Visualization as a means to explain the mechanisms of artificial intelligence is an emerging field. The guiding principle behind AI user interfaces is not to replace but augment and expand users' abilities [329].

As stated in [330], “the core idea of conventional visualization libraries is that visualizations are compositional by nature. They provide many components (circles, squares) for visualization as well as procedures for building more complex forms”. Unfortunately, most traditional visualization tools such as D3.js, Vega and Tableau do not have specialized capabilities for visualizing feature spaces, neural network layers or iterative design space exploration [331] during model development.

Visualization and interpretability instruments can be divided into two main categories: (a) model-agnostic instruments which are not tailored for any single model and (b) specialist visualization instruments which are targeted at certain models. Most instruments in the first category are called “explainable AI” (XAI). Previous XAI libraries used visual representations to explain attributes embedded inside the machine learning models. However, most libraries nowadays focus on visualizing important elements of neural network (embeddings or attention maps) [332]. Examples of the former are many kinds of line and bar charts used for explanation (feature importance diagrams). Two advanced methods to explain models based on machine learning are: LIME (denoting Local Interpretable Model-agnostic Explanations) and SHAP (denoting SHAPley Additive exPlanations). SHAP provides visualization facilities for summary and dependency graphs and is basis for newer tools but both have been shown to be vulnerable to adversarial attacks [333]. ELIS [334] is a modern replacement for SHAP that is being used to explain BERT forecasts. The explosion in Transformer models is one of the reasons why the toolbox keeps growing so fast. Visualization techniques for Transformers [331] can be divided into two categories (i) focused: visualizations on specific aspect (e.g., attention), ii) holistic: visualization systems which explain whole model or its development process .

Here are some examples of use visualization and explanatory tools to look at data and models created in this study.

One way to deal with arithmetic data sets is to transform the EEG-based algorithmic complexity time series (arithmetic test data) into Gramian Angular Fields or Recurrence Plots. These graphs show how the system changes as it moves through different regions of the value

space over some period of time (usually length of the time series). The shapes look like pictures (e.g., Figure 7.16(a)) so we put these images into a CNN for classification. Figure 7.16 shows a picture taken before testing and its evaluation and representation, while Figure 7.17 shows a picture obtained during testing. Both are shown here: the original picture and the one modified several times in the neural network.

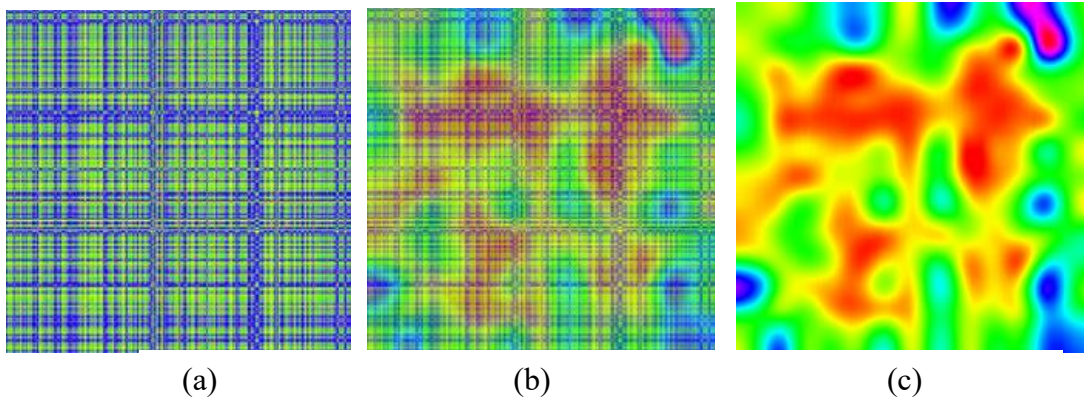


Figure 7.16. (a) An original “pretest” image; (b) and (c) the heatmap for one of the first layers and 9th layer (using VGG-16 model) when input into a CNN for classification.

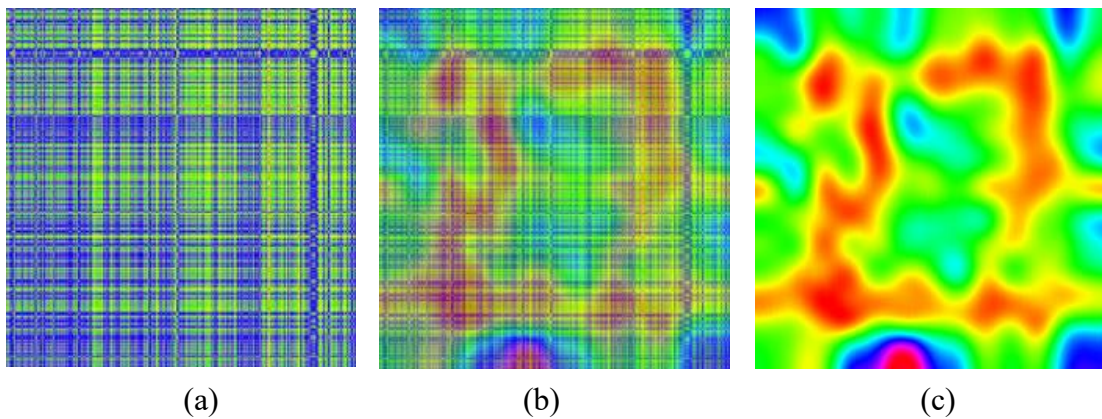


Figure 7.17. An original “during-the-test” image; (b) and (c) the heatmap for one of the first layers and 9th layer (using VGG-16 model) when input into a CNN for classification.

The pictures are from different groups and were chosen to have some resemblance. The illustrations in the preceding pictures, especially (c), help with the discrimination problem.

Figure 7.18 shows heatmaps showing an alternative version. They show how AC values are allocated chronologically for a 50 step part of these sequences.

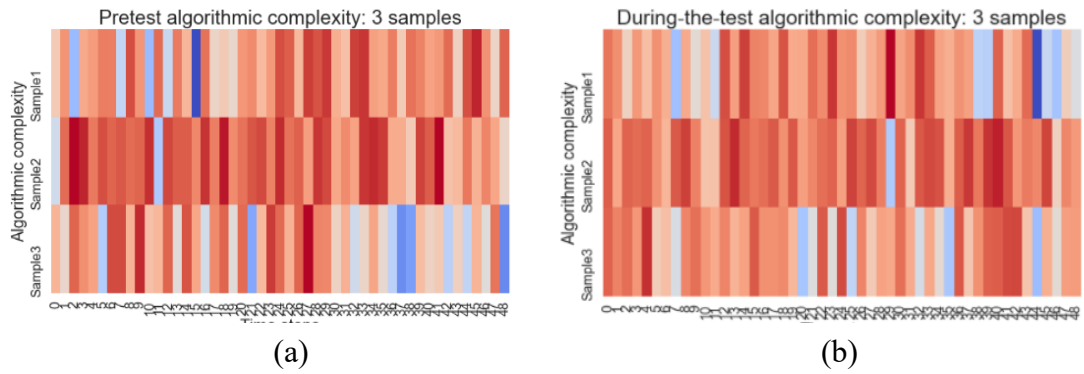


Figure 7.18. Heat maps of AC (algorithmic complexity) for six instances of AC time series from different categories (a) three “pretest” instances and (b) three “during-the-test” instances. Hues represent different levels of AC values.

This kind of visualization is not directly useful for discrimination but lays the foundation for attention maps which are used to explain a model (e.g., Figure 7.12).

Below are shown some basic tools: distribution density line plots (Figure 7.19), partial dependence plots (Figure 7.20), SHAP summary plots (Figure 7.21).

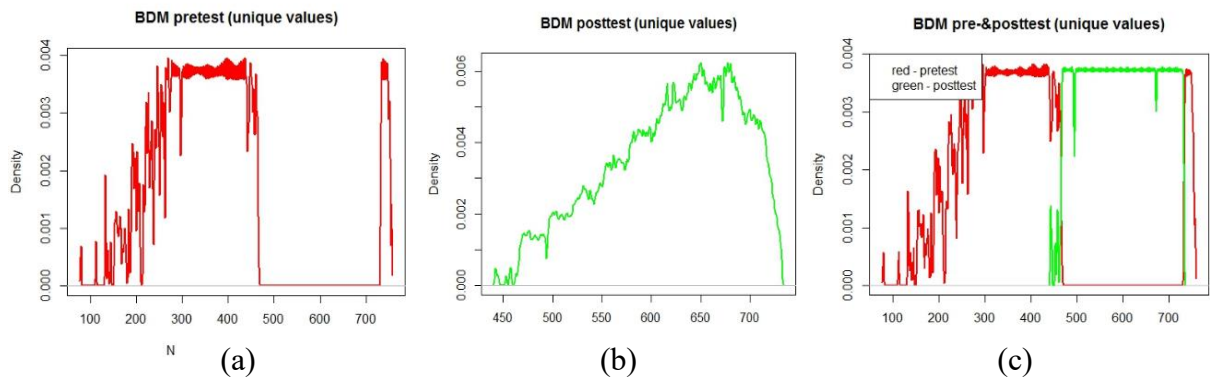


Figure 7.19. Distribution density graphs showing allocation of AC levels (after conversion of AC values to text form back): (a) density of distinct words in “pretest”, (b) during-the-test dataset and intersection of these two graphs (c). [292]

The graphs in Figure 7.19 show how much the unique word collections (or AC values) from “pretest” category (a) and “during the test” category (b) are different, with a bit overlap at the center (c).

7.3.2. Models based on sepsis data

The following plots are examples of explainability/interpretability of the “Sepsis” models with comments.

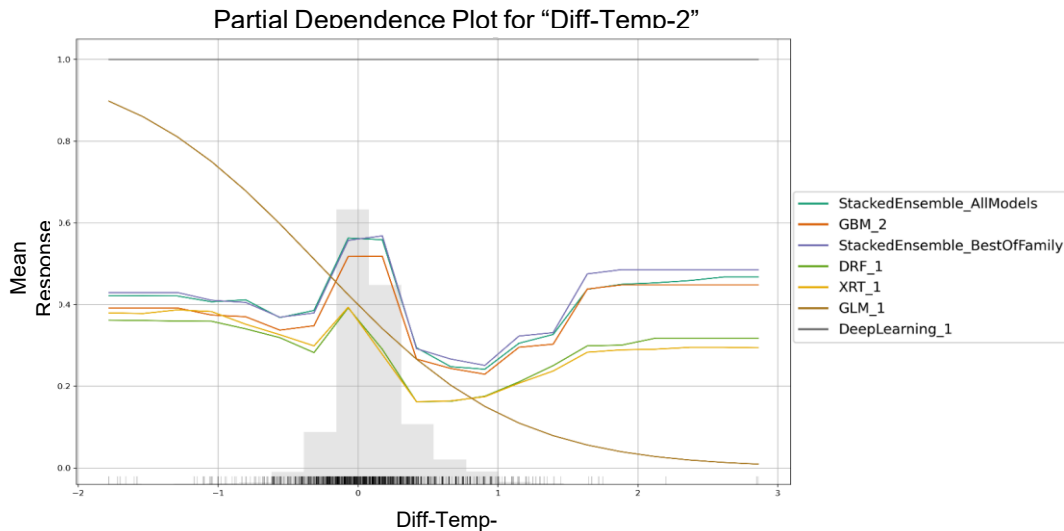


Figure 7.20. Partial dependence visualization shows body temperature derivative across several machine learning models: Stacked Ensembles, Gradient Boosting Machine, Distributed Random Forest, Extreme Randomized Trees, Generalized Linear Model and Deep Learning.

Partial dependence plot (PDP) is used to identify the most influential variable(s) on the response variable. Temperature derivative has highest predictive significance within 0.1–0.2 degrees interval (see Figure 7.20). The amount of observations represented by the bars at the base of the diagram shows that this trend is generally present: after 0.4 degrees the significance of the derivative of temperature decreases drastically, but then goes back up again and peaks at about 1.7 degrees. The statement flips over for a negative value (mirror it as if seen in a reflection). This trend is similar for most models except GLM (Generalized Linear Model): GLM is based on linear regression so the biggest significance is given to the negative value (-2 degrees), but these events are rare in our dataset (they decrease in significance as values get positive). The SHAP-summary figure (Figure 7.21) shows the relationship between feature importance (Y axis: higher is more important) The positions of traits on X axis show how much they affect predictions (positions to right of “0” are for sepsis, left ones are for a lower risk of sepsis etc.; further from “0” the effect is stronger). Hue corresponds to quantitative value of attribute (from minimum to maximum): so if body temperature increases dramatically (fast ascent) then this is a sign that sepsis risk is low and if there’s only a little change in temperature it could be because immune system of sepsis patient isn’t able to generate an adequate response to infection (insufficient fever reaction).

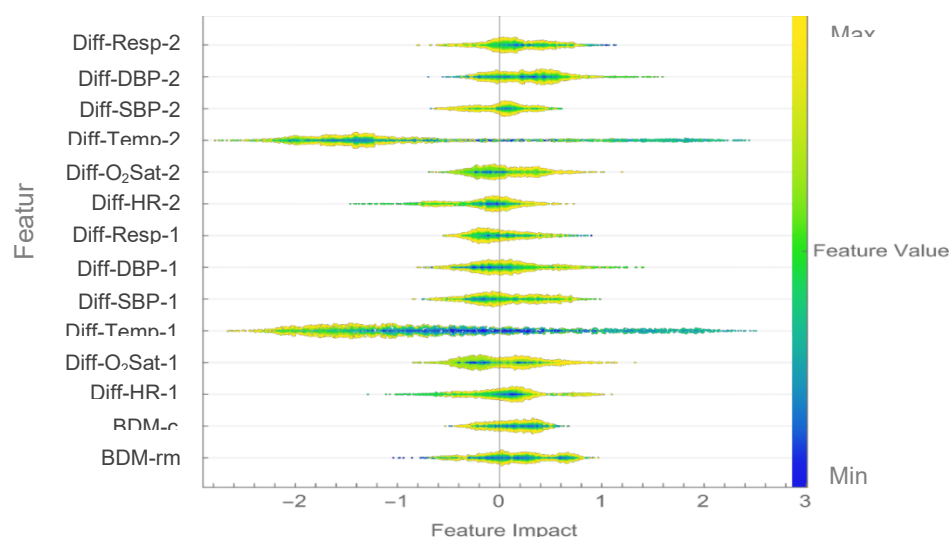


Figure 7.21 A SHAP-summary plot generated with Wolfram Engine 13.3 for one of the GBM models built on sepsis data.

Figure 7.21 shows a very clear example of the algorithmic complexity projected from respiratory and metabolic characteristics (BDM-rm). The algorithmic complexity is positively related to sepsis risk (yellow and green dots are increased values). Low number of blue dots means low risk of sepsis.

Table 7.8 shows two samples (one sepsis and other non-sepsis) along with the SHAP values numerically represented.

Table 7.8. SHAP values for two samples

Class\ Parameters	BDM rm	BDM c	Diff HR 1	Diff O ₂ Sat 1	Diff Temp 1	Diff SBP 1	Diff DBP 1	Diff Resp 1	Diff HR 2	Diff O ₂ Sat 2	Diff Temp 2	Diff SBP 2	Diff DBP 2	Diff Resp 2
Class\Para- meters	1.01	0.97	1.05	0.86	1.21	1.84	1.50	1.15	1.45	0.92	0.61	1.00	1.27	1.18
Non-sepsis	0.99	1.03	0.96	1.17	0.82	0.54	0.67	0.87	0.69	1.08	1.65	1.00	0.79	0.85

Note: BDM-rm and BDM-c are KCC values; rest of the columns are difference in value for six parameters of interest over three hours.

7.4. Considerations regarding the practical application of generated models

Epilepsy

The new Kolmogorov-Chaitin complexity estimation method for EEG analysis can be demonstrated on models trained on an epileptic dataset.

There are also hurdles that could slow epilepsy seizure prediction deployment (beyond more research): start with EEG registration in a non-clinical setting (wearable electrodes and continuous signal recording). This study's trials also showed that Kolmogorov-Chaitin complexity estimate of brain created linguistic complexity is person specific so model training would have to be patient specific.

When the two individual EEG clip collections were combined to get a shared word embedding model, classification experiments showed that Word2Vec (trained on the combined data set) and classifier built on data from first person (measured by AUC) performed well (1.00 from 1st individual, 0.98 from 2nd). When tested on dog 2 test set "combined" classifier did as in Table 7.4 but dog 1 test set had AUC of 0.51.

But TF-IDF model was built differently: original dog 2 data split into training and test sets. Five of 42 preictal clips and 59 interictal clips were tested; novel data generation procedure used only for training. With this setup good classification performance of the model suggests it could be applied to all prospective cases. Even with small number of training/testing samples (especially pre-epileptic) an appropriate data generation approach would give an effective model. This is individualized medicine and may guide future study.

Sepsis

A severe infection, sepsis can cause organ malfunction and life-threatening complications. Sepsis data models are more realistic than epilepsy data models and ready for software application incorporation, fine-tuning and clinical implementation in a specific intensive care unit but there are things to keep in mind.

First of all, before using a prediction model in clinical practice it must be independently externally validated for generalizability across cohorts [335]. Most newly developed risk prediction algorithms perform poorly on external samples [336]. The sepsis risk prediction model in this work was tested on patients from another hospital (set B in the original data set) or externally validated. Balanced group of 9993 samples including 4975 sepsis samples had AUC = 0.84 (95% CI 0.833–0.847).

Retraining the prediction model with target medical center datasets improves prediction accuracy [337]. When there is a limited dataset with the same purpose transfer learning improves model performance in target medical institution [338]. The interpretability of machine learning models is how well their decision-making process can be understood and used in clinical practice. Most prediction models cannot be implemented because they are uninterpretable [339, 340]. This was discussed in Section 7.3.

So, two approaches are proposed to address the performance problem in this case: (a) build a new model with recent data and use insights from similar models in previous research (i.e., replicate a prior study with new data); (b) transfer learning to train newly built model on new data.

Future work could follow various paths: studying EEG research; testing sepsis prediction system developed in this study in Moldova (if needed to improve it using transfer learning with local data); observing how ICU specialist uses it in real life.

7.5. Software applications developed using established models

The study successfully embedded the most effective models into five software applications. Two of them (including GUI) are shown below.

Algorithmic Complexity (by BDM & NLP/Word2Vec model) EEG Analyser



Figure 7.22. Graphical User Interface (GUI) to analyze the algorithmic complexity of EEG using ISAAC-EEG-BDM-NLP/Word2vec_2.1.0 software.

One application uses KCC and NLP/Word2vec model (Figure 7.22) to analyze EEG data. Users can import EEG segment in original ".mat " format, convert it into ".csv" format for further processing and do some transformations on it (e.g. dividing it into sub segments so that it can be visualized easily). The application allows you to display the traditional 16 channel EEG of selected sub segment; any particular channel separately or calculate dynamics of KCC value over time; visualize the " WordCloud " to represent most common words (transformed from KCC values into words); use bar chart to display most frequent words. Last step is to show 25 vectors for each of 10 most common words in analyzed segment. These actions can be done by clicking on the buttons in the left panel. Finally, the text forecasts samples chosen beforehand (GBM model built on Word2Vec) and displays the result as either an interictal (normal) or preictal (pre-epileptic) clip. The upper horizontal bar menu “How to use this application” describes recommendations for using the application. The ISAAC-EEG-BDM-NLP/Word2vec app is available online at https://viapascurta.shinyapps.io/ISAAC_EEG_NLP_2_1/.

The sepsis prediction system (Figure 7.23) has been incorporated into a Shiny application Sepsis-ISAAC-new-2_1_2 which is also available online at: https://viapascurta.shinyapps.io/Sepsis_ISAAC_new_2_1/.

The “ISAAC Sepsis Expert – 2.1.2” app was designed for practical use by any physician interested in sepsis. The left panel (Fig. 7.23) allows you to enter the hourly values of six physiological parameters over three hours, then in the main panel as a table these data are displayed. Once the data is correct it is sent for processing (button) and approved by the model embedded in the application (pressing button below). If you press the button below you will get an assessment result high risk sepsis or low risk sepsis. To input a new set of data (a new case) you have to reset the application using the button on top left panel. In top bar menu button How to use the application gives instructions how to use the application.

ISAAC Sepsis Expert - 2.1.2

New data input

Case/Patient ID

patient 001

Patient diagnosis

Gangrene

Parameters value for last 3 hours

1 hour ago

Heart Rate

69

SaO2

100

Tempature

36.6

Systolic BP

98

Diastolic BP

81

Respir. Rate

17

2 hours ago

Heart Rate

62

SaO2

100

Tempature

36.6

Systolic BP

92

Diastolic BP

47

Respir. Rate

15

3 hours ago

Heart Rate

71

SaO2

97

Tempature

37.7

Systolic BP

105

Diastolic BP

54

Respir. Rate

13

Display data

Powered by ISAAC-MED:



Expert

How to use this application

patient 001 : Gangrene

Entered data:

	Timing	HR	O2Sat	Temp	SBP	DBP	Resp
1	-1 hour	69	100	36.6	98	81	17
2	-2 hours	62	100	36.6	92	47	15
3	-3 hours	71	97	37.7	105	54	13

Verify and submit data

Sepsis Risk Evaluation

Get evaluation results

Low Sepsis Risk

Figure 7.23. GUI of software application for early prediction of sepsis (ISAAC Sepsis Expert - 2.1.2).

There are two other R/Shiny software applications also: The first one for EEG analysis can be found at https://viapascurta.shinyapps.io/ISAAC_EEG_1_2_2/. It is similar to the first application for EEG analysis but uses text analysis tools (TF-IDF approach). The second one is a demo version of the sepsis prediction tool available at: https://viapascurta.shinyapps.io/ISAAC_sepsis_demo/. A desktop application written in Python using Wave package was also developed for use in a specific ICU. Its functions These models are similar to those described above only more comprehensive. The GUI and source code of these software applications can be found in Appendix 4.

7.6. Concluding discussion and future steps

Models documented in the literature exemplify state-of-the-art ML models. Sepsis case is better defined since it is included in the latest International Guidelines for the Management of Sepsis and Septic Shock. Epilepsy case has been summarized based on high performance results found in literature.

In both fields an inherent problem is that number of cases, patient databases, research methodology and reporting format used by each study vary.

Epilepsy

Researchers use various algorithms and methods for EEG classification (see Table 7.9). According to their usual use models may be divided into two types: (a) to detect the occurred seizures; (b) to predict the seizures.

Table 7.9. Comparative performance of models developed on the same EEG data set

Metrics\ References	2014 Kaggle Competition (the first 10 results)	Usman S. et al. (2021)	Ibrahim A.K. et al. (2023)	Created system (based on CBOW Word2vec model) (95% CI)
ROC AUC	0.78 – 0.82	-	-	0.94 – dog 1 0.97 – dog 2
Recall	-	0.94	0.81 ± 0.02 (dog 1) 1.00 (dog2)	1.00 (0.86 - 1.00) - dog 1 0.98 (0.87-1.00) – dog 2
Specificity	-	0.96	-	0.91 (0.79 - 0.98) – dog 1 0.96 (0.86-1.00) – dog 2
Accuracy	-	-	-	0.94 (0.86 - 0.98) – dog 1 0.97 (0.91-0.99) – dog 2
False Positive Rate	-	-	0.08 ± 0.02 (dog1) 0.04 ± 0.03 (dog2)	0.082 – dog 1 0.038 – dog 2
Positive Likelihood Ratio	-	-	-	11.25 (4.41-28.67) – dog 1 24.4 (6.27-94.97) – dog 2
Negative Likelihood Ratio	-	-	-	0.00 – dog 1 0.02 (0.00-0.17) – dog 2
Positive Predictive Value	-	-	-	0.86 (0.67-0.96) – dog 1 0.95 (0.84-0.99) – dog 2
Negative predictive Value	-	-	-	1.0 (0.91-1.00) – dog 1 0.98 (0.89-1.00) – dog 2

Note: ROC AUC – area under the ROC curve, CI – confidence interval

Only two papers use the same data as in our study (Usman S. et al., 2021 [341] and Ibrahim A.K. et al., May 2023 [342]. Table 7.9 can be used to compare models described in these two papers with the best model in our current work (CBOW Word2vec model [343]) and those presented in Kaggle competition [51], from which data for present research have been taken. Performance of models has improved over time. Comparing models in detail is difficult since different performance metrics are being used but it seems that the models described in last three columns (2021, 2023 and one developed by present research) perform well.

Sepsis

In this specific context [344] we find that (a) the definition of sepsis is not consistent across studies; (b) data sources, data preprocessing methods, machine learning models used, feature engineering techniques and types of inclusion criteria differ greatly among studies; (c) AUROC value increases with time to sepsis onset; (d) AUROC increase is mainly due to the use of ML as a tool for feature engineering; (e) deep neural networks combined with Sepsis-3 diagnostic criteria perform better when applied to time series data from sepsis patients. Evaluation criteria and reporting standards facilitate development of machine learning models for clinical applications.

There are many models in the medical literature that predict clinical outcomes [320]. Models can be classified into two groups: those which estimate likelihood of a specific outcome (e.g., diagnosis) and those that anticipate if a given outcome (e.g., event) will occur in the future. Traditionally, prediction models (hereinafter “models”) have been developed with regression-based techniques. Short-term outcomes have commonly been predicted by logistic regression while long term outcomes have usually been predicted by cox regression [345]. Many reviews have commented that studies which describe how a prediction model was made and checked (changes included) often leave out important details that would help readers judge methods and get a full, clear and accurate picture of how accurate the model is at making predictions and other details [319, 346].

Table 7.10 shows several systems that have performed well in the classification and prediction of sepsis. The literature has documented these systems over the past five years.

Table 7.10. High-performance systems for sepsis prediction (last 5 years)

Nr.	Author(s)	Publication Year	AUC ROC/ accuracy*	Number of features/algorithm
1.	Siri D. et al.	2025	0.99	WoOA selected features / GPT-GNN
2.	Dalal S. et al.	2024	0.96-0.99	Laboratory data (18) / DL (RNN, transformer)
3.	Ke L. et al.	2023	0.93-0.99	11 differentially expressed genes/SVM
4.	Laxmi L. et al.	2023	0.94*	44/ Stacked Ensembles (DT, SVM, LR, KNN, NB)
5.	Sadasivun S. et al.	2022	0.93*	ECG + demographic features / RF
6.	Misra D. et al.	2021	0.948	15 / RF
7.	Wickramaratne D., Mahmud S.	2020	0.97	36 / BiLSTM
8.	Kok C. et al.	2020	0.98	40/ TCN

Table 7.10. Continued

9.	Helguera-Repetto, A. et al.	2020	0.944	25 / MLP
----	-----------------------------	------	-------	----------

Note: AUC ROC – area under the ROC curve, WoOA - wolverine optimization algorithm, GPT-GNN—generative pre-training graph neural network, RNN – recurrent neural network, SVM – support vector machine, DT – decision tree, LR – logistic regression, KNN – k -nearest neighbor, NB – naïve Bayes, ECG – electrocardiogram, RF – random forest, BiLSTM – bidirectional long short-term memory, TCN – temporal convolutional network, MLP – multi-layer perceptron.

As stated above, it is hard to compare such systems. For example, for the highest-performing sepsis prediction systems (Wickramaratne D., Mahmud S., and Kok C. et al.), it is due to insufficient available information: study design and, in particular, validation details and system 9 is only for newborn children; all systems use a different number of cases from few hundreds to several thousand); Dalal S. et al system is for non ICU patients while Siri D. et al is predicting mortality and not sepsis onset.

These uncertainties can be partially addressed by referring to a more reliable source for comparison (International Guidelines for the Management of Sepsis and Septic Shock November 2021 [274]). This document describes how machine learning could be used to improve screening tools performance. That conclusion is based on a meta-analysis of seven studies with 42,623 patients who wanted to predict nosocomial sepsis. The analysis showed AUC

We also cite two papers for the adoption of our findings into literature: The first paper called our work “the first studies using BDM for the analysis of physiological signals” [347]. The second one is presenting us approach to estimate KCC on biomedical data as an example of “Estimations of Integrated Information Based on Algorithmic Complexity and Dynamic Querying” on biological models [348].

Stage conclusions and future work:

- Representation learning tools can have great benefits but their impact varies according to the type of data they are applied to.
- This methodology and tools can give a lot of insight about a particular dataset which makes it possible to understand better what is in the data; to reduce complexity; achieve better results (e.g., text classification).
- Several techniques used in this area (stemming, lemmatization and sub-words) seem less appropriate for the ‘artificial’ type of text we have here.
- The best model for EEG analysis of our data set was Word2Vec model (Continuous Bag-of-Words CBOW version) which is highly recommended for future work.

- The Block Decomposition Method when used to estimate algorithmic complexity has properties of a data representation/dimension reduction technique too: e.g. it compresses original 16 channel EEG time series into univariate time series (7.67 MB -> 250 KB for a 10 min EEG clip). This compression still contains information needed for classification.
- The sepsis prediction system developed on the initial data set performs well enough so that it can be applied practically in clinical settings but systems developed on EEG data sets need further research and refinement of these models. Nevertheless the results are conservative estimates of what high performance might be expected from the system.
- The high performance of the sepsis prediction system makes it suitable for use in clinical settings in Republic of Moldova. The system can help patients as ICU doctors will be warned about imminent risk of sepsis; special literature has been published and documented that antibiotic treatment was started early due to this warning and the outcomes were favorable.

7.7. Synthesis of Chapter 7

In this chapter we apply ML techniques on preprocessed data for predicting sepsis and EEG signals (complexity science approach) and start by stating standard stages of developing ML systems (Figure 7.1). Main elements and steps in creating an ML system are shown in Figure 7.1, while Figure 7.2 shows data partitioning into training and test sets. For medical applications the chapter mentions TRIPOD-AI for transparent reporting of rules and PROBAST-AI for bias assessment as well as metrics to evaluate model performance (TPR, TNR, accuracy, precision, recall, ROC analysis, precision-recall curves, confusion matrices, variable importance) and special role of Kolmogorov-Chaitin complexity (KCC) metrics from algorithmic information dynamics.

The chapter then moves on to exploratory data analysis and ML with transformed EEG time series. Algorithmic complexity (AC) visualizations aggregated over windows show differences between interictal (normal) and preictal (pre-epileptic) EEG clips in Figure 7.3 where preictal clips have higher AC values at the beginning with narrower fluctuations. EEG data is converted into two formats: numerical vectors based on TF-IDF (Table 7.1) and text that works with NLP (Table 7.2 where KCC values are turned into words). Different ML models are applied. The TF-IDF approach with GBT (Gradient Boosted Trees) has a diagnostic accuracy of 0.92 on test sets after data augmentation and thresholding (Figure 7.4 ROC curve and confusion matrix). Word2Vec models parameterized as in Table 7.3 also perform well (AUCs on Dog 1 and Dog 2 test sets are 0.944 and 0.966 respectively for Skip-Gram in Figures 7.5 and 7.6), CBOW vs. Skip-Gram comparison in Figure 7.7 shows no difference between them. Accuracy around 0.85–0.90 is

achieved by FastText (Figure 7.8), BERT with 768 hidden sizes and 12 attention heads gets AUC 0.670 but about 8% higher with denoising from Word2Vec (Figures 7.9 and 7.10). Attention maps in BERT internals are shown in Figures 7.11 and 7.12. BiLSTM models in PyTorch and Keras versions stabilize after epochs (Figure 7.13) with accuracies of 0.891 and above 0.980 in training (Figure 7.14). Table 7.4 shows the best NLP models; for Dog 2 Word2Vec (CBOW) at AUC 0.968 is best.

For sepsis data transformed into vectors based on BDM estimated AC on physiological matrices and parameter dynamics (see Table 7.12 earlier), four ML algorithms are trained GLM, GBM, DRF and DNN with optimal hyperparameters in Table 7.5; on a test set of 2881 samples (892 sepsis, 1989 non-sepsis) GBM has AUC 0.953, recall 0.942, accuracy 0.957 (ROC curves in Figure 7.15 and detailed metrics in Table 7.6). In Table 7.7 we compare against winners of the 2019 Challenge (AUC 0.83–0.87) and guidelines from 2021 (AUC 0.89), AUC 0.95, accuracy 0.96. On external validation on another hospital's data showed an AUC of 0.84.

Explainability is achieved through visualizations for EEG hierarchical gradient heatmaps (Figures 7.16 and 7.17) AC distribution heatmaps (Figure 7.18) density plots (Figure 7.19) class overlaps; sepsis partial dependence plots (Figure 7.20) temperature derivative impact SHAP summary plots (Figure 7.21) with values (Table 7.8) feature contributions (e.g. higher BDM-rm favors sepsis).

EEG models require patient-specific training since each brain has its own “language”, sepsis models are clinical ready (ISAAC-EEG-BDM-NLP/Word2vec for EEG analysis; ISAAC Sepsis Expert for predictions, Figure 7.23). In the final sections we compare the models to literature: Table 7.9 EEG (created CBOW Word2Vec AUC 0.94–0.97 vs priors) and Table 7.10 recent sepsis systems (AUC 0.93–0.99).

In summary (a) representation learning and KCC estimation in this work fundamentally reduce dimension and pattern capture of biomedical signals (e.g., 7.67 MB EEG clip size reduced to 250 KB for a 10 min clip), preserve classification utility while treating complexity as proxy of processes running in brain; (b) GBM-based sepsis prediction system is high clinical readiness with superior performance metrics (95.3% by AUC) so early interventions can be done in intensive care unit (ICU); hence it is suitable for implementation in clinical settings in Republic of Moldova ; (c) EEG analysis methods based on NLP on KCC-transformed data can predict seizure with models like Word2Vec (e.g. CBOW version of Continuous Bag-of-Words was found to be the best model for EEG analysis of the data set in this research showing over 96% by AUC performance) but need personalization and validation for practical use e.g. notifying patient about imminent epileptic seizure not to do anything until danger subsides.

8. LARGE LANGUAGE MODELS AND MULTI-AGENT SYSTEMS IN INTENSIVE CARE: SEPSIS MANAGEMENT IMPROVEMENT

8.1. Foundations of Large Language Models, Retrieval-Augmented Generation and Multi-Agent Systems in Intensive Care

AI in healthcare (especially ICU) has transformed clinical decision making, constant surveillance of patients and therapeutics. LLMs are neural networks with state-of-the-art capabilities trained on huge text corpora to understand natural language, to produce coherent text and to reason in an effective way. Examples include GPT-5, Gemini and biomedical versions, such as Palmyra Med. Transformers are used to handle sequential data to produce contextually appropriate and clinically relevant outputs. LLMs have a great potential to synthesize medical information, predict outcomes, and coordinate across multidisciplinary teams in the intensive care setting where rapid, evidence-based decisions are needed for life-threatening conditions such as sepsis (an organ dysfunction-driven, life-threatening condition that results from a dysregulated host response to infection).

Mathematically, LLMs are built on top of the transformer model proposed by Vaswani et al [293]. Self-attention computes the relative importance of different parts of a sequence. The self-attention operation is defined as::

$$Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \quad (8.1)$$

Q and K are query key and value matrices from input embeddings respectively; d_k is the dimension of the keys. This allows LLMs to learn long range dependencies needed for EHRs (narrative notes, vital sign trends, lab results) interpretation as they contain information about past events. LLMs can be fine tuned on domain specific repositories such as MIMIC-IV database [349] to predict sepsis onset from time series data for intensive care applications. RNN with attention has been incorporated in LLM frameworks to model temporal sequences where h_t at time step t is updated via

$$h_t = \tanh(W_h[h_{t-1}; x_t] + b_h), \quad (8.2)$$

attention layers integrate past information to improve prediction accuracy. Domain adapted models like Palmyra Med pretrained on large biomedical corpora (PubMed abstracts, clinical guidelines) are more suitable for applications like differential diagnosis in sepsis where conditions such as ARDS overlap with presentation patterns.

However, LLMs have limitations: they can hallucinate (generate plausible but factually incorrect information), cannot access knowledge beyond the training cutoff or proprietary sources, etc. These issues are particularly important in intensive care where clinical guidelines change very fast

(e.g. Surviving Sepsis Campaign [274] every few months) and patient data is very heterogeneous. Retrieval augmented generation (RAG) addresses these limitations by dynamically retrieving relevant external information to augment the LLM prompt. Typical RAG pipeline consists of a retriever (usually dense vector search engine) and a generator (LLM itself). The retriever (e.g. BERT [350]) projects queries and documents into a common vector space and ranks them by cosine similarity:

$$\cos(\theta) = \frac{\mathbf{q} \cdot \mathbf{d}}{\|\mathbf{q}\| \cdot \|\mathbf{d}\|}, \quad (8.3)$$

where q and d are vectors of dimension n , $\|q\|$ and $\|d\|$ represents the Euclidean norm. RAG significantly alleviates the parametric burden on LLMs by offloading factual knowledge to external databases, and enhances the factual accuracy of sepsis management (e.g., antibiotics recommended based on local resistance patterns extracted from surveillance data published by European Center for Disease Prevention and Control (ECDC) [351]).

This idea is extended in Multi-agent (MA) systems, where complex clinical tasks are decomposed into collaborative specialized agents that communicate through message passing protocols. These are typically formulated as MDPs where each agent's policy, $\pi(s)$ chooses an action a based on the current state s to maximize a common reward function. In ICU, MA systems can be organized as multidisciplinary teams (agents for diagnostics, therapy planning and guideline verification), similar to actor-critic RL [352]. Recently, Cemri et al. [353] summarized failure modes of MA LLM systems using MAST taxonomy, which includes specification problems (FM-1.1/FM-1.2), inter-agent misalignment (FM-2.3/FM-2.4) and verification failures (FM-3.1/FM-3.3). Such failures can have fatal consequences in sepsis where every hour delay in antibiotic administration increases mortality by $\sim 7.6\%$ [354]: specification errors may not comply with Surviving Sepsis Campaign one-hour bundle, misalignment delays source control in abdominal sepsis, verification failures may terminate therapy prematurely without confirming microbial de-escalation which then accelerates antimicrobial resistance. Tested empirically > 200 traces across 7 MA frameworks Failure rates are high Design principles and scalable verification pipelines needed (e.g., authors' LLM-as-a-Judge).

Computationally, MA systems can be seen as Dec-POMDPs [355], where agents act on local observations of a global state while joint policies maximize expected discounted reward.

$$V(\pi) = E \left[\sum_{t=0}^{\infty} \gamma^t r_t | \pi \right], \quad (8.4)$$

where γ is the discount factor and r is the reward. In sepsis applications rewards could be defined to penalize delays in fluid resuscitation or vasopressor initiation based on SOFA scores. The inference engines (LLMs such as Palmyra Med) are specialized for this purpose: RAG-augmented prompts are processed to reduce hallucinations. For example, agents can extract EHR segments from vector databases like FAISS [356] using domain specific embeddings generated by BioBERT [357].

The intensive care is also data-heavy, which further motivates their use. ICUs generate terabytes of heterogeneous data on a daily basis from monitors, ventilators and lab systems (often overwhelming clinical staff). LLMs can extract knowledge from this data (e.g. to predict septic shock progression with LSTM based time series forecasting models [358] by feeding sequences of vital signs $x_1, \dots, x_T$ to generate risk probabilities). RAG adds external instructions with up-to-date data. MA systems distribute computational tasks among agents (e.g. Bayesian networks for probabilistic inference [359] or Q-learning for treatment optimization [360]).

Other challenges include compliance with data privacy regulations (e.g., HIPAA), which can be addressed by federated learning approaches where model gradients are aggregated but raw patient data are not centralized [360] and ethical issues (LLM outputs could have biases that widen sepsis outcomes disparities for underrepresented populations) [361]. Fairness constraints during the model training [362] are among the mitigation techniques.

Sepsis prediction research using LLM augmented methods can outperform qSOFA based scoring systems (gradient-boosting machine models AUROC 0.953 [278]). RAG also enables real time integration of literature on biomarkers (procalcitonin) [363]. MA systems need orchestration for scaling up computationally. AutoGen [364] allows agents to communicate over APIs with latency optimized by parallel processing. In ICUs where teams work under high reliability principles [365], MA designs need to have clear hierarchies and communication protocols to avoid misalignments — for sepsis this means that agents ought to prioritize time sensitive actions (e.g. blood cultures before antibiotics) [274]. MA LLM systems are still young, but early pilots in antimicrobial stewardship [366] show promising paths for wider deployment, possibly combined with hybrid LLM symbolic AI frameworks that enforce logical constraints from pathophysiological models.

LLMs are powerful generators; RAG is for factual grounding; MA architectures enable collaborative reasoning. Combined, these technologies constitute a synergistic triad that can propel precision medicine in the intensive care setting, provided that systemic deficiencies identified in the literature are systematically addressed through design evaluation and iteration.

8.2. Using Retrieval-Augmented Generation for Domain Specific Tasks

We start out with a non-clinical case study to demonstrate the applicability of RAG to domain specific tasks and thereby establish the methodological foundation that can be applied later on medical settings (e.g., sepsis management).

8.2.1. Case Study: Retrieval Augmented Generation for Domain-Specific Question Answering

8.2.1.1. Background and Study Context

Recent progress in NLP is due to large language models (LLMs) like GPT, Gemini and Claude that are very good at producing human-like answers to a variety of tasks after being trained on huge text corpora. However, they cannot reason over domain knowledge outside of their training data. For instance, in philosophy accurate answers must refer to passages from particular books or manuscripts.

Our pilot investigation [367] reported here constructed a RAG-based question answering application for the domain of social philosophy and philosophy of management. The external knowledge source was a published book with 50 self-evaluation questions in Russian text (original text translated to English for system building; generated answers back-translated to Russian for expert review). The main purpose of the project was to show if an efficient RAG pipeline for domain specific queries can be created by using free open source tools and to verify the accuracy and consistency of the answers. In 2024 we used LangChain [368], ChromaDB [369], LlamaIndex [370], GPT-3.5 Turbo [371], UMAP embedding visualization [310] and Triad evaluation framework [372]. Overall RAG architecture is a standard retrieve-and-generate as shown in Figure 8.1.

8.2.1.2. Problem Definition

The main problem is that LLMs have a fixed knowledge cutoff, making them unsuitable for fields like philosophy of management where loyalty to source texts (not general knowledge) is crucial – if you need to discuss “temperology” or conflicts between theories in philosophy, you must go straight to the source text. LLMs hallucinate content (make up plausible but incorrect stuff) without augmentation.

The study was designed to test two hypotheses: 1) whether a fully working RAG system could be constructed using open-source components to answer questions taken from a philosophical text database and 2) how accurate/consistent the answers were versus the original material. Challenges in implementation included processing abstract philosophical prose,

capturing subtle semantic nuances in embeddings, and achieving computational efficiency without compromising quality.

8.2.1.3. Methods and Implementation

The RAG system (Figure 8.1 from author’s original paper [367]) was developed to retrieve passages from the external book at inference time and insert them into the LLM prompt. The pipeline consisted of three main steps: vector store building, retrieval, and response generation. Constructing a vector store As shown in Figure 8.2 vector store construction was loading the book documents, chunking the text (e.g., into 256-character segments with no overlap using SentenceTransformerTextSplitter), embedding it with models like all-MiniLM-L6-v2 or BAAI/bge-small-en-v1.5 and indexing the vectors into a vector database.

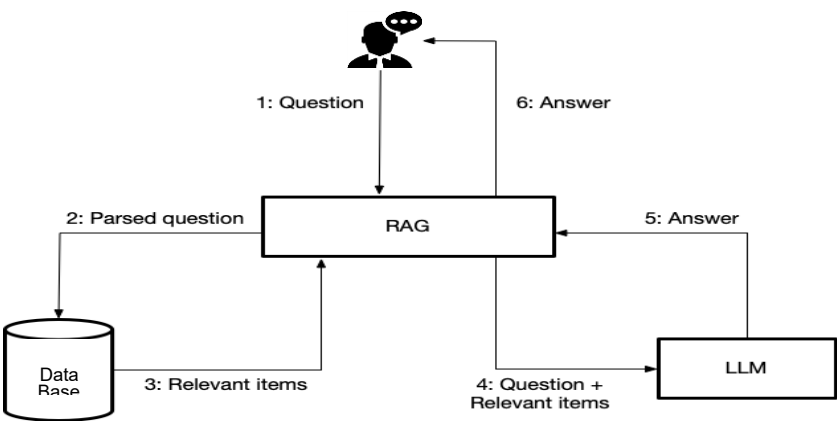


Figure 8.1. Architecture of the created RAG system. [367]

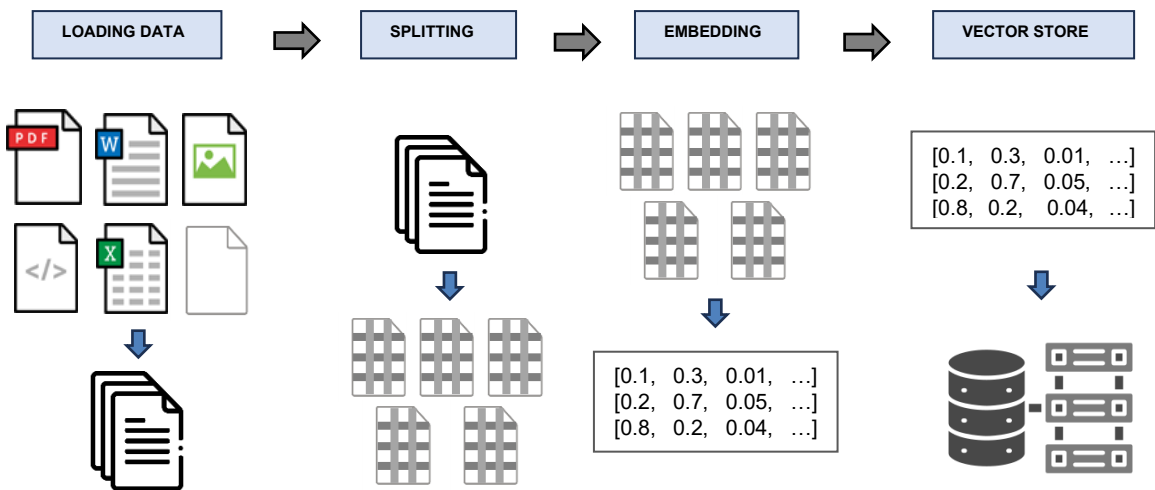


Figure 8.2. Process of building the vector database. [367]

Three variants of RAG were constructed and compared: (1) LlamaIndex Sentence Window Retrieval that returns the most relevant sentence with its surrounding contextual window; (2) Cosine Similarity ChromaDB-based retrieval after embedding fixed size chunks; (3) LlamaIndex Auto-Merging Retrieval that hierarchically organizes the text and retrieves parent nodes to provide broader context when child nodes match query. Figure 8.3 depicts the retrieval and generation steps.

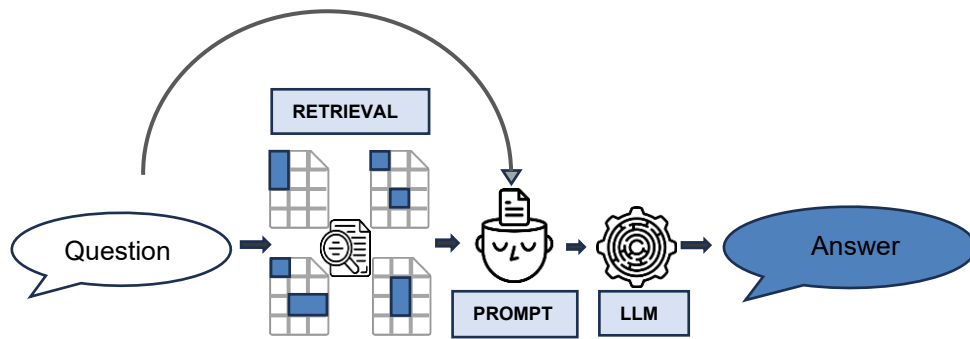


Figure 8.3. Main steps of the RAG data pipeline. [367]

User questions were embedded and compared to the vector store using top-k cosine similarity and the chunks obtained from this search were included in the prompt sent to GPT-3.5 Turbo with explicit instructions to only use that context.

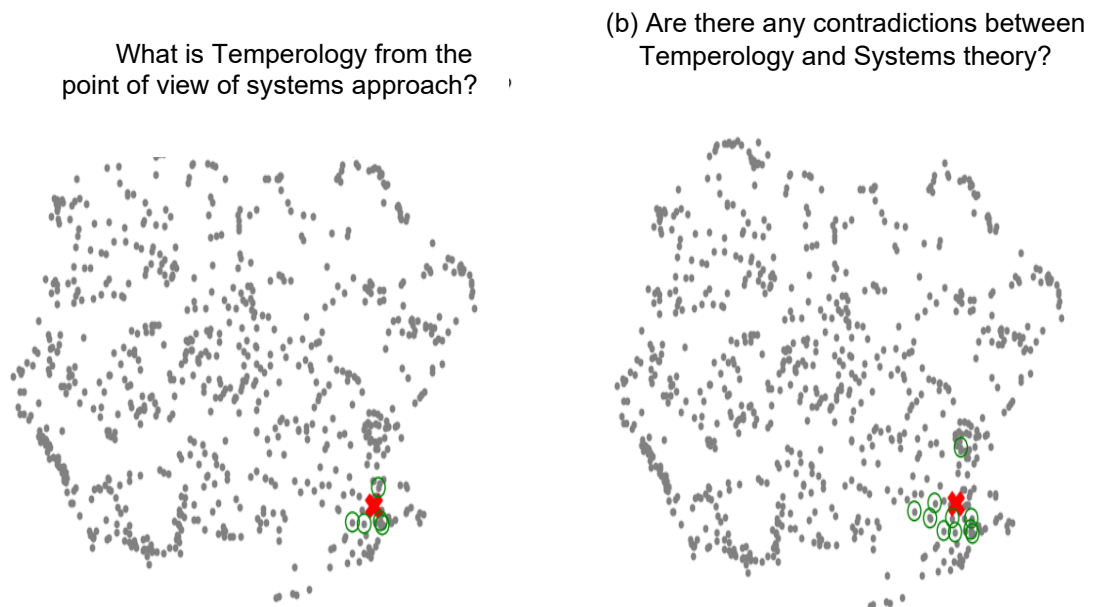


Figure 8.4. Query embedding proximity to retrieved chunks for relevant questions.

[367]

How does Mars influence the well-being of humans?

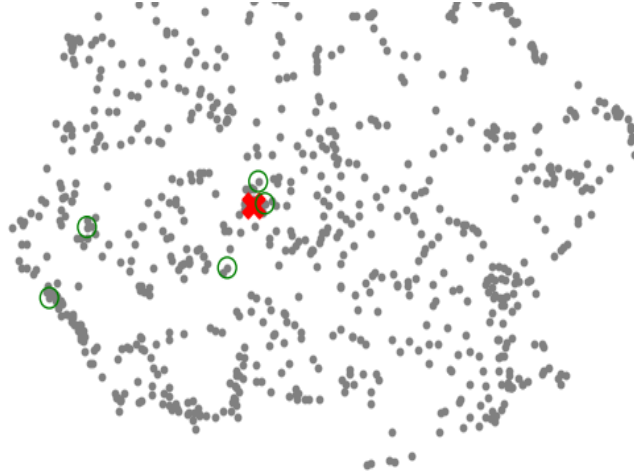


Figure 8.5. Query embedding and spread of retrieved chunks for an irrelevant query. [367]

We used UMAP for dimensionality reduction and visualization of embedding quality (Figure 8.4 shows the proximity of a relevant query (red cross) to retrieved data points (green circles); Figure 8.5 shows distribution obtained for irrelevant query). The Triad evaluation framework was used to measure context relevance (Figure 8.6), groundedness (Figure 8.7) and answer relevance (Figure 8.8) supported by Chain-of-Thought reasoning traces.

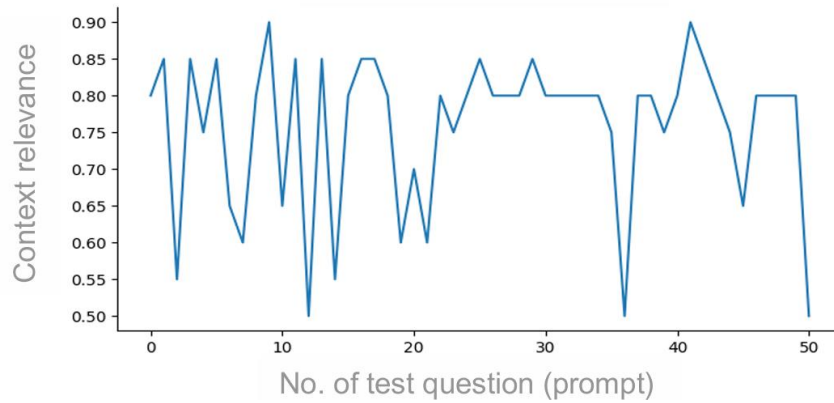


Figure 8.6. Evaluation of context relevance using Chain-of-Thought reasoning. [367]

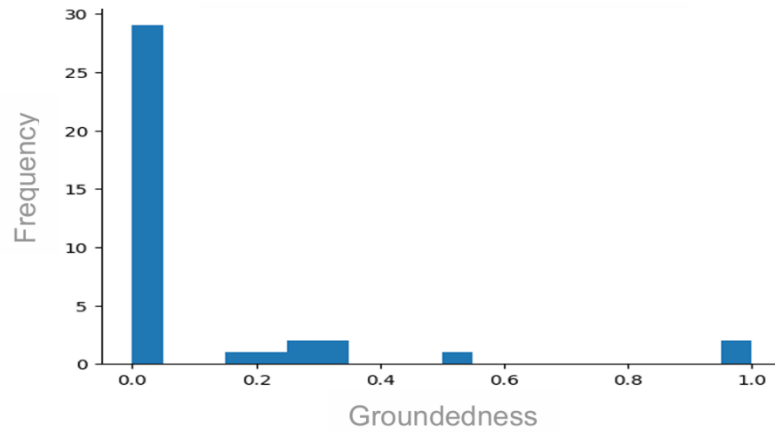


Figure 8.7. Evaluation of groundedness using Chain-of-Thought reasoning.[367]

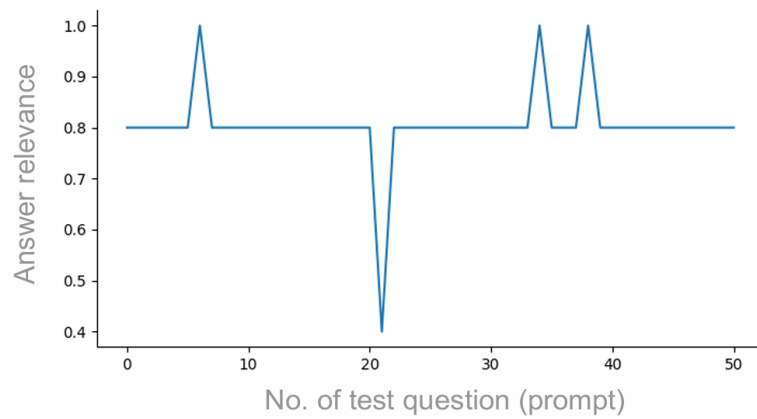


Figure 8.8. Evaluation of answer relevance. [367]

8.2.1.4. Study Results

Table 8.1 shows the results of the three RAG variants for each of the metrics (answer relevance, context relevance, groundedness, latency and computational cost). The ChromaDB-based variant had the best balance between high relevance scores with low latency. For the example query “How many elements are there in temperological structure of personality?” both RAG variants correctly answered “three elements”, much better than unaugmented ChatGPT baseline. Independent experts rated the answers as acceptable overall.

Table 8.1. The performance of the three RAG variants [367]

Application	Answer relevance	Context relevance	Groundedness	Latency (seconds)	Cost (USD)
RAG based on Sentence Window Retrieval	0.804	0.760	0.107	6.10	0.003
RAG based on ChromaDB	0.897	0.865	0.240	3.10	0.002

Table 8.1. Continued

RAG based on the auto-merging approach	0.846	0.529	0.621	37.84	0.006
--	-------	-------	-------	-------	-------

8.2.1.5. Discussion and Takeaways

Overall, the ChromaDB variant performed better, suggesting that cosine similarity is a good choice for abstract philosophical texts. Some configurations show lower groundedness, indicating residual LLM drift that can be mitigated using hybrid retrieval-generation techniques. The small size of the dataset limited the scope of this study, as did translation difficulties and the absence of real-time data integration. However, the strengths of this study show that open-source RAG pipelines are indeed feasible to use, and UMAP visualizations have diagnostic value. Results show that RAG is generalizable to specific domains (retrieving sepsis guidelines and literature in intensive care units) and provide a transferable methodological template.

8.3. RAG in ICU

Retrieval augmented generation is the breakthrough that bridges the gap between generative strengths of large language models and the stringent demands for accurate, up to date knowledge in medicine and intensive care: by combining retrieval with LLM generation (i.e. adding a retrieval mechanism to LLM generation), RAG provides external context dynamically so hallucination is substantially reduced and static training data are overcome.

In clinical practice, RAG enhances diagnostic accuracy, treatment recommendations, and evidence synthesis. The basic idea is to index source documents in a vector database, embed queries and chunks of docs into vectors (usually using Sentence-BERT [373]), and rank them by cosine similarity (top k results go into LLM prompt). Naive RAG (retrieve-and-generate), Advanced RAG (pre- and post-retrieval optimizations like sliding windows) and Modular RAG (more flexible, reconfigurable pipelines) are common variants.

In the intensive care unit, where patient data and guidelines are constantly changing, RAG is especially helpful in sepsis management, as it can draw information from electronic health records (EHRs) or PubMed to inform decisions about fluid resuscitation, vasopressor administration, and antimicrobial selection. Multimodal embeddings address the heterogeneity of sources; federated RAG architectures [374] alleviate privacy concerns. Fast indexing solutions such as FAISS [356] can meet latency requirements and achieve real time ICU decision support. Hybrid sparse-dense retrieval methods improve precision in the presence of noisy clinical records.

Overall RAG provides the basis for later integration into multi-agent systems (grounded knowledge that agents can use collectively).

8.4. Multi-Agent AI Systems for Sepsis Care in ICUs

Sepsis remains one of the leading causes of morbidity and mortality in intensive care units, resulting from a dysregulated host response to infection that quickly progresses to life-threatening organ dysfunction (20–30 % global mortality in severe cases). Clinicians are constantly faced with diagnostic uncertainty, increasing antimicrobial resistance and the need to adhere to constantly updated guidelines such as Surviving Sepsis Campaign. For each hour of delay in appropriate antibiotic therapy, mortality risk increases by 7–9 %.

LLMs are good at NLP but hallucinate and cannot reliably retrieve up-to-date domain knowledge. RAG directly addresses these limitations: outputs are anchored in literature and guidelines retrieved via vector-based similarity search; multi-agent systems, which decompose subtasks into specialist agents, further enhance reliability (similar to ICU teams collaborating in frameworks like CrewAI [377]).

8.4.1. Case Study: Proof-of-Concept Multi-Agent System for Early Sepsis Management in Community-Acquired Pneumonia

8.4.1.1. Problem Definition

Empiric antibiotics and adherence to guidelines in early sepsis management are often complicated by waiting for culture results [375]. In this proof-of-concept study, we tested a multi-agent system developed to assist intensivists with a representative case of community-acquired pneumonia sepsis from the MIMIC-IV database (Case ID 10020786): an 86-year-old woman with chronic obstructive pulmonary disease who presented with hypotension (BP 103/49 mmHg), hypoxia (SpO₂ 90%), and marked leukocytosis (WBC 41,100/mm³) [375, 376].

8.4.1.2. Methods

The system consisted of three specialized agents orchestrated with CrewAI and powered by the Palmyra-Med 70B LLM (Sepsis Management Agent for overall strategy; Antibiotic Recommendation Agent for empiric therapy selection; Guidelines Compliance Agent to ensure alignment against 2021 Surviving Sepsis Campaign recommendations). RAG was performed using Chroma vector databases populated with recent sepsis literature and guidelines with cosine similarity retrieval (Figure 8.9). The architecture and data flow are shown in Figure 8.10. The input data were structured as a clinical vignette (Table 8.2).

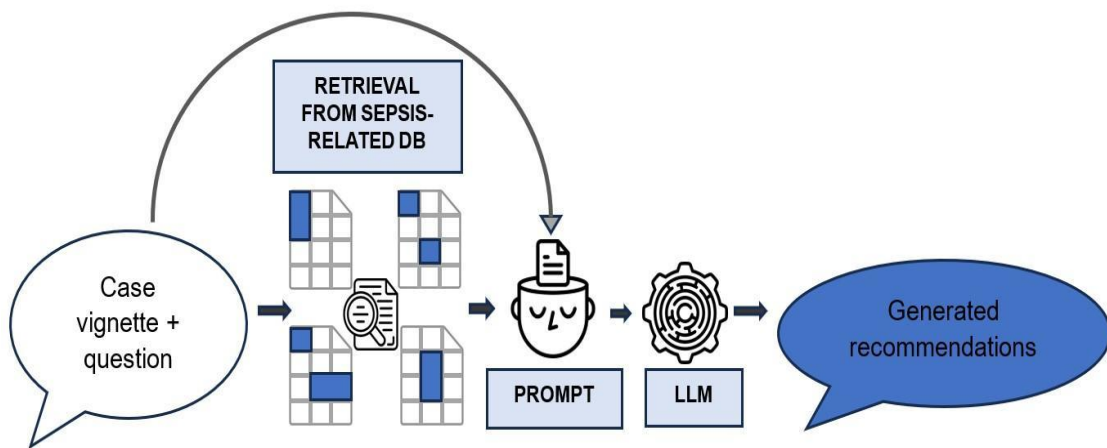


Figure 8.9. RAG component in the multi-agent sepsis system.

The architecture (Figure 8.10) takes vignettes and fetches data that is relevant for LLM generation, before the generation happens. Testing was carried out on a cloud-based application [375].

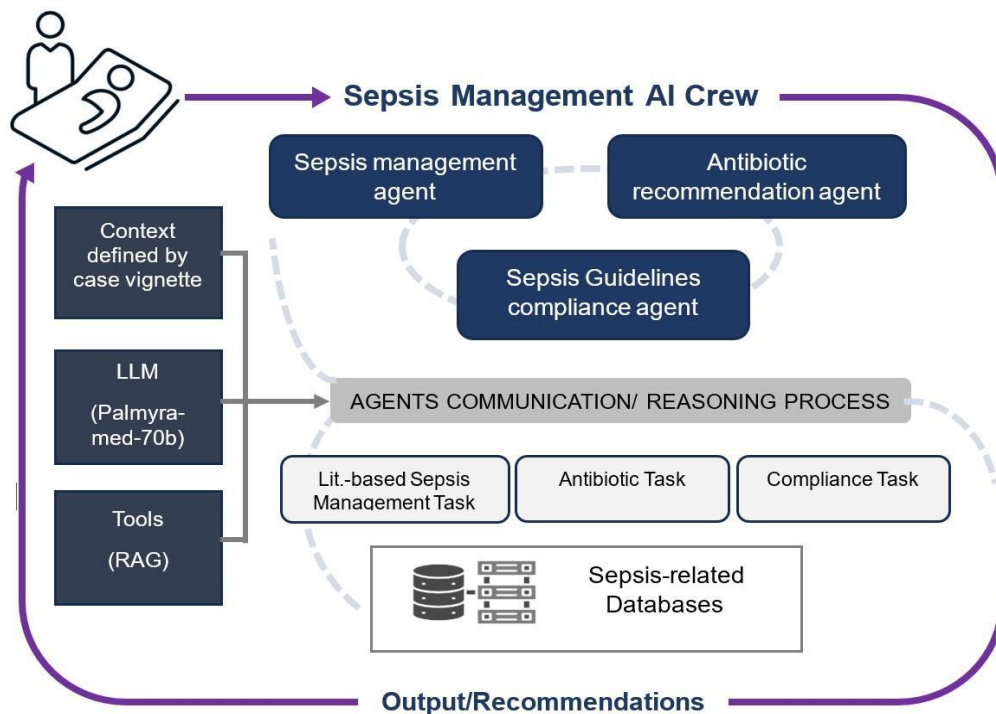


Figure 8.10. Overall architecture and data flow of the multi-agent system. [376]

The vignette (Table 8.2) gives structured input to the agents.

Table 8.2. Clinical vignette used for the pneumonia sepsis case [376]

Vignette section	Description
Diagnosis	Sepsis due to pneumonia
Demographic	female, 86 years
Vital signs	HR 76 bpm, BP 103/49 mmHg, RR 22 rpm, blood O ₂ saturation 90%, temperature 37.2°C
Laboratory findings	WBC 41.100/mm ³ (41.1 x 10 ⁹ cells/L), blood glucose 225 mg/dL (12.5 mmol/L), BUN 13 mg/dL (0.72 mmol/L)
Medical history	Chronic obstructive pulmonary disease, atrial fibrillation, type 2 diabetes mellitus without complications, hypothyroidism
Special comments	Community-acquired pulmonary infection

8.4.1.3. Results

The agents worked together to generate recommendations including broad-spectrum antibiotics, fluid resuscitation, close monitoring, and strict adherence to guidelines (Fig. 8.11). Evaluation of the TruLens framework (using GPT-3.5 Turbo) showed that the answers were relevant but the groundedness scores were low (Figure 8.12). Two intensivists evaluated the outputs as acceptable (Cohen's $\kappa = 0.622$, $p = 0.003$).

(a)	Literature-based sepsis management recommendations Early administration of antibiotics is crucial in sepsis management, especially in patients with suspected infections and systemic inflammatory response syndrome criteria. In the case of sepsis due to pneumonia in an 86-year-old female with multiple comorbidities, including COPD and diabetes, prompt antibiotic therapy is essential. Considering the patient's condition, a broad-spectrum antibiotic covering typical pathogens in pneumonia should be initiated promptly. Close monitoring for clinical response and adjusting therapy based on culture results is recommended. Additionally, supportive care to stabilize vital signs and manage comorbidities is vital in improving outcomes for this patient. Regular reassessment and collaboration with a multidisciplinary team are key in providing comprehensive care for this complex case.
(b)	Antibiotic recommendations Tailoring empiric therapy for pulmonary infections, particularly Community-Acquired Pneumonia (CAP), should involve considering local microbial resistance patterns and individual patient factors. For CAP, initial empiric therapy may include a combination of a beta-lactam antibiotic like ceftriaxone or cefotaxime with a macrolide such as azithromycin. Alternatively, monotherapy with a respiratory fluoroquinolone like levofloxacin can be considered, especially for patients with beta-lactam allergies. In cases where there is a risk of MRSA or <i>Pseudomonas aeruginosa</i>, adding vancomycin or linezolid for MRSA and piperacillin-tazobactam, cefepime, or meropenem for <i>Pseudomonas</i> coverage may be necessary. It is crucial to adjust therapy based on local antimicrobial resistance data to ensure effective treatment.
(c)	Compliance with sepsis and septic shock management guidelines (2021) Prompt initiation of broad-spectrum antibiotics in sepsis is crucial, but de-escalation based on clinical improvement and culture results is recommended to prevent unnecessary therapy. Tailoring empiric therapy for pneumonia should consider local resistance patterns and patient factors to ensure effective treatment. To achieve compliance, consider the following suggestions: To enhance compliance with recommendations, ensure prompt initiation of broad-spectrum antibiotics for suspected pneumonia in the elderly patient with comorbidities. Tailor therapy based on local resistance patterns, consider beta-lactam/macrolide combination or fluoroquinolone for Community-Acquired Pneumonia. Adjust treatment according to culture results and collaborate with a multidisciplinary team for comprehensive care.

Figure 8.11. Recommendations generated by MA system: a) sepsis management recommendations, b) antibiotic recommendations and c) compliance with guidelines statement. [376]

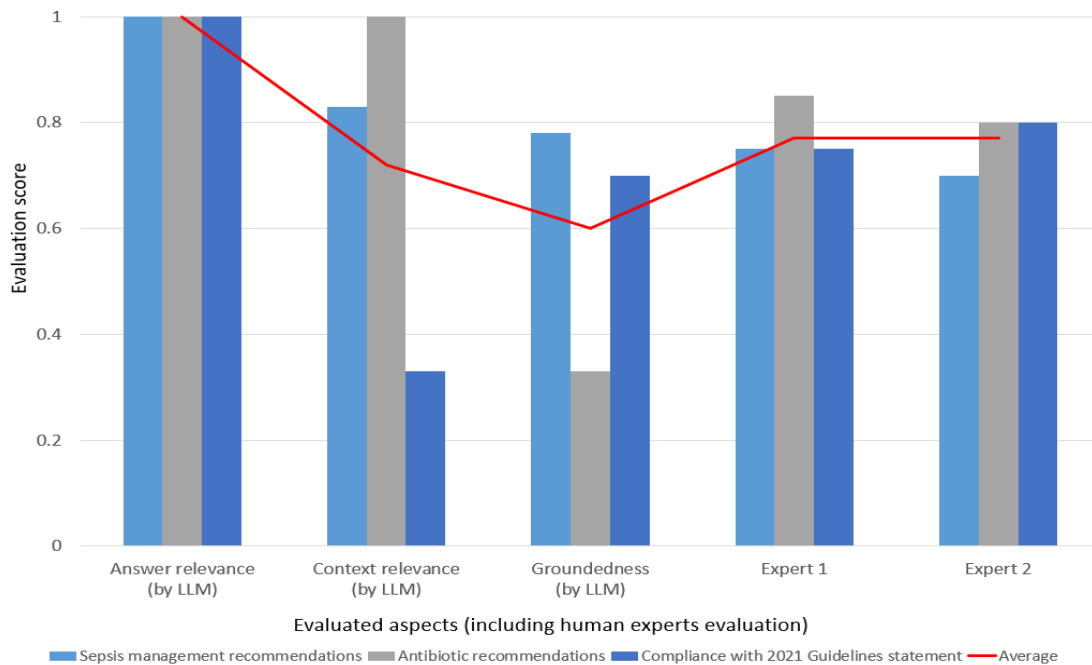


Figure 8.12. TruLens evaluation results for the single-case proof-of-concept. [376]

8.4.1.4. Discussion and Lessons Learned

The system was useful in early sepsis but to avoid hallucinations outputs were grounded in evidence so the negligible agreement between LLM and experts shows this needs to be done quickly. Single case; vignette-based approach Future: broader validation, real-time integration.

This proof of concept demonstrates the validity of MA-RAG in sepsis and paves the way for extensions.

8.4.2. Follow-up Case Study: Scaling Multi-Agent System to Multiple Sepsis Cases and Hallucination Mitigation

8.4.2.1. Problem Statement

Based on the single case study, the paper examined the scalability and hallucination mitigation of the system for ten presentations of sepsis (eight MIMIC-IV and two from literature) including necrotizing fasciitis, genitourinary sepsis, peritonitis and ventilator-associated pneumonia [378, 379].

8.4.2.2. Methods

We kept the same three-agent architecture and RAG pipeline but expanded Chroma vector stores to include more literature (see Figure 8.13 for agent level data flows).

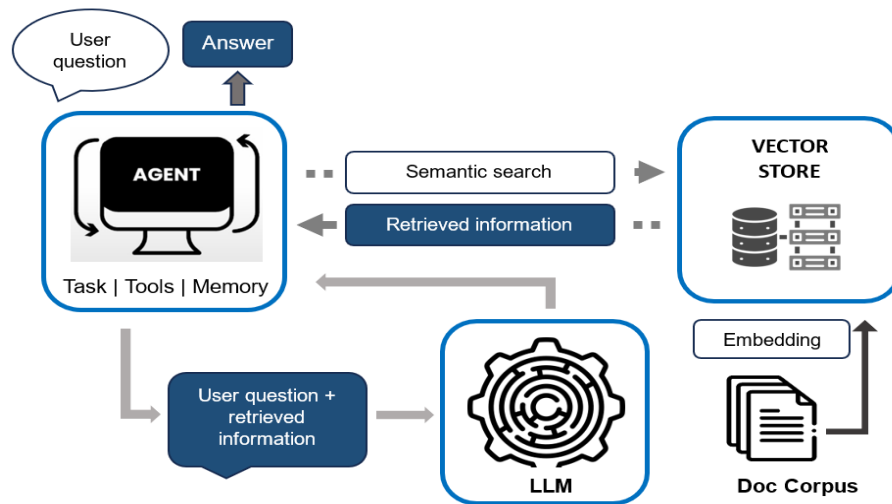


Figure 8.13. Individual agent architecture and internal data flows. [378]

8.4.2.3. Results

Comprehensive assessment [376,378,379] using TruLens metrics and dual expert human review. Guidelines were generally followed in recommendations (e.g., vancomycin + piperacillin-tazobactam along with urgent surgical source control in necrotizing fasciitis case ID 10015860). However, groundedness scores varied across outputs, with occasional hallucinations (e.g., unsupported statements of altered mental status) observed (see Table 8.3 for inputs/outputs, Table 8.4 for TruLens results, and Figure 8.14 for programmatic versus human scores.).

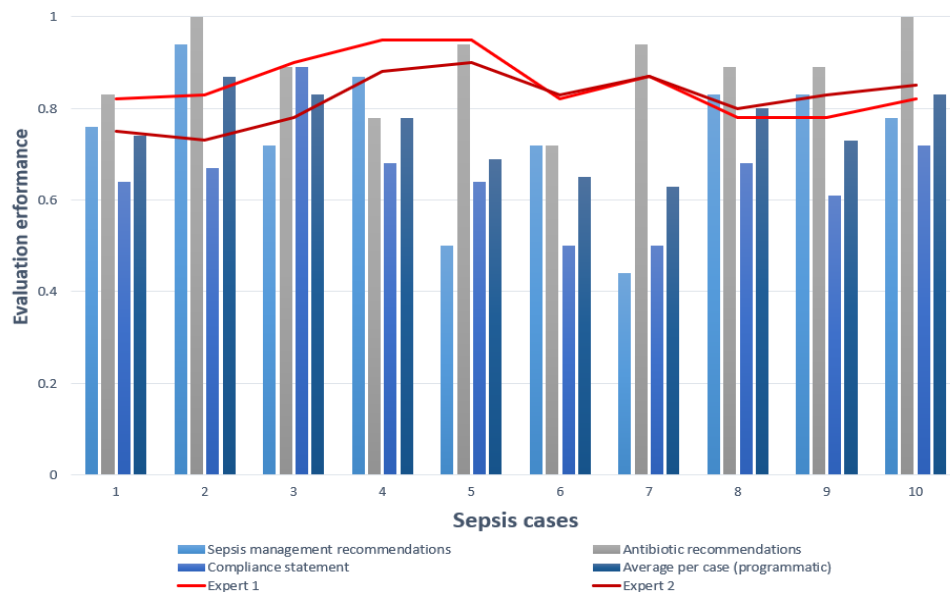


Figure 8.14. Comparison of programmatic (TruLens) and human expert evaluation scores.

[378]

Tables 8.3 and 8.4 provide sample inputs/outputs (Table 8.3) and detailed evaluation results (Table 8.4) by TruLens. Adapted from [378, 379, 398].

Table 8.3. Input vignettes and generated outputs for selected sepsis cases

Sepsis case/aspects	Case # 1 (ID 10015860)	Case # 2 (ID 10020786)	Case # 3 (ID 10003400)
Clinical vignette of the case	<i>Diagnosis:</i> Sepsis due to necrotizing fasciitis of the foot; <i>Demographics:</i> male, 53 years; <i>Vital signs:</i> HR 106 bpm, BP 116/62 mmHg, RR 23 rpm, blood O₂ saturation 93%, temperature 38°C; <i>Laboratory findings:</i> WBC 18,000/mm³ (18.0 x 10⁹/L), blood glucose 215 mg/dL (11.9 mmol/L), blood creatinine 0.8 mg/dL (70.7 µmol/L), BUN 16 mg/dL (5.7 mmol/L), blood lactate 0.7 mmol/L <i>Medical history:</i> Type 2 diabetes mellitus with diabetic chronic kidney disease, diabetic neuropathy, ulcer of the foot, acquired absence of right toe; <i>Special comments:</i> Skin and soft tissue infection	<i>Diagnosis:</i> Sepsis due to community-acquired pneumonia; <i>Demographics:</i> female, 86 years; <i>Vital signs:</i> HR 76 bpm, BP 103/49 mmHg, RR 22 rpm, blood O₂ saturation 90%, temperature 37.2°C; <i>Laboratory findings:</i> WBC 41,100/mm³ (41.1 x 10⁹/L), blood glucose 225 mg/dL (12.5 mmol/L), BUN 13 mg/dL (4.6 mmol/L); <i>Medical history:</i> Chronic obstructive pulmonary disease, atrial fibrillation, type 2 diabetes mellitus without complications, hypothyroidism; <i>Special comments:</i> Pulmonary infection, community-acquired pneumonia	<i>Diagnosis:</i> Sepsis of genitourinary origin; <i>Demographics:</i> female, 72 years; <i>Vital signs:</i> HR 110 bpm, MAP 55 mmHg, RR 22 rpm, blood O₂ saturation 99%, temperature 37.7°C; <i>Laboratory findings:</i> WBC 4,600/mm³ (4.6 x 10⁹/L), blood glucose 129 mg/dL (7.2 mmol/L), blood creatinine 1.4 mg/dL (123.8 µmol/L), BUN 30 mg/dL (10.7 mmol/L), , red blood cells 2.480.000/mm³ (2.48 x 10¹²/L), hemoglobin 7.6 g/dL (76 g/L), platelet count 57.000/mm³ (57 x 10⁹/L), blood lactate 3.3 mmol/L; <i>Medical history:</i> Hypertensive chronic kidney disease, Malignant neoplasm of rectum, Multiple myeloma, Atrial fibrillation, Chronic congestive heart failure, Obesity; <i>Special comments:</i> Genitourinary infection
Sepsis management recommendations	The management recommendations for sepsis in a patient with necrotizing fasciitis of the foot should include prompt initiation of appropriate antibiotics targeting the likely pathogens causing the infection. Source control through surgical intervention to remove infected tissue is crucial. Given the presence of sepsis criteria such as <i>altered mental status</i>*, tachypnea, leukocytosis, hyperglycemia, and <i>hyperlactatemia</i>*, close monitoring of vital signs, fluid resuscitation, and vasopressor support if needed are essential. Considering the patient's comorbidities like diabetes and chronic kidney disease, careful fluid management and monitoring of kidney function are important. Additionally, addressing the foot ulcer and ensuring	In managing sepsis due to community-acquired pneumonia in an 86-year-old female with multiple comorbidities, it is crucial to promptly administer appropriate antibiotics based on the suspected infection source. Given the patient's vital signs and laboratory results, initial resuscitation with fluid therapy and oxygen support is essential. Considering the patient's age and comorbidities, a tailored antibiotic regimen targeting common pathogens in community-acquired pneumonia should be initiated promptly. Close monitoring of the patient's response to treatment, including vital signs, oxygen saturation, and laboratory parameters, is crucial. Additionally, early involvement of a multidisciplinary team for	For the management of sepsis in a 72-year-old female with genitourinary infection and multiple comorbidities, initial steps should include rapid resuscitation to address shock. This involves ensuring adequate fluid resuscitation, initiating appropriate antibiotics early, and optimizing hemodynamics. Specific goals include maintaining a MAP of at least 65 mmHg, central venous pressure of 8-12 mmHg, urine output of at least 0.5 mL/kg/h, and a mixed venous oxygen saturation of at least 70%. Given the patient's clinical parameters, close monitoring for organ dysfunction and perfusion abnormalities is crucial. Additionally, considering the patient's

	proper wound care to prevent further complications is vital. Regular assessment for signs of septic shock and organ dysfunction is necessary for timely intervention. Collaborating with a multidisciplinary team including infectious disease specialists, nephrologists, and wound care experts is recommended for comprehensive management.	comprehensive care and consideration of the patient's overall clinical status and goals of care are important in optimizing outcomes for this complex case	chronic conditions like chronic kidney disease and malignancies, a multidisciplinary approach involving specialists from various fields would be beneficial to tailor the management plan accordingly
Antibiotic recommendations	Recommendations for skin and soft tissue infections should be based on local resistance patterns and adjusted according to the clinical situation. For necrotizing fasciitis, consider multidrug therapy with vancomycin or linezolid along with piperacillin-tazobactam, a carbapenem, or ceftriaxone and metronidazole. For non-purulent cellulitis/erysipelas, initial empiric therapy could include vancomycin and piperacillin-tazobactam. In cases of severe purulent infections, consider using vancomycin, daptomycin, linezolid, telavancin, or ceftaroline. It is important to consider urgent surgical consultation and adjust antibiotic choices based on the specific clinical scenario and local resistance patterns.	For community-acquired pneumonia, initial empiric therapy should include a combination of a beta-lactam antibiotic such as ampicillin + sulbactam, ceftriaxone, or cefotaxime along with a macrolide antibiotic like azithromycin or clarithromycin. Alternatively, monotherapy with a respiratory fluoroquinolone such as levofloxacin or moxifloxacin can be considered. In cases where there are risk factors for MRSA or Pseudomonas aeruginosa, adding vancomycin or linezolid for MRSA and piperacillin-tazobactam, cefepime, or meropenem for Pseudomonas coverage may be necessary. It is important to tailor the antibiotic choices based on local resistance patterns and consider the patient's individual clinical condition.	Consider using Ceftriaxone as the initial empiric therapy for Acute Pyelonephritis based on the IDSA criteria. Trimethoprim-sulfamethoxazole can also be effective against a broad range of urinary pathogens, but its use should be avoided if local resistance rates are high. Ciprofloxacin is another option that is effective against many Gram-negative organisms, including Pseudomonas. For severe cases, intravenous options like fluoroquinolones, aminoglycosides, or extended-spectrum cephalosporins should be considered. It is important to monitor local resistance patterns and avoid fluoroquinolones if resistance rates exceed 10%. In cases of severe penicillin allergies, carbapenems or alternative agents based on susceptibility data should be considered.
Compliance statement	The recommendations do not comply as they lack specific guidance on de-escalation of antibiotics, duration of therapy, and the importance of prompt removal of intravascular access devices in sepsis management, which are crucial aspects highlighted in the guidelines. To achieve compliance, consider the following suggestions: To ensure compliance with the recommendations, consider tailoring antibiotic therapy to local resistance patterns, adjusting based on clinical scenario and involving a multidisciplinary team for comprehensive care. Regularly assess for septic shock signs and organ dysfunction, prioritize source control through surgical intervention, closely monitor vital signs, fluid resuscitation, and consider vasopressor support if needed.	Prompt initiation of broad-spectrum antibiotics in sepsis is crucial, but de-escalation based on clinical improvement and culture results is recommended to prevent unnecessary therapy. Tailoring empiric therapy for pneumonia should consider local resistance patterns and patient factors to ensure effective treatment. To achieve compliance, consider the following suggestions: To enhance compliance with recommendations, ensure prompt initiation of broad-spectrum antibiotics for suspected pneumonia in the elderly patient with comorbidities. Tailor therapy based on local resistance patterns, consider beta-lactam/macrolide combination or fluoroquinolone for Community-Acquired Pneumonia. Adjust treatment according to culture	The recommendations do not comply because they do not align with the specific steps and goals outlined for managing sepsis in a 72-year-old female with genitourinary infection and multiple comorbidities. To achieve compliance, consider the following suggestions: Suggested steps: Ensure rapid resuscitation with adequate fluid, early antibiotics, and optimized hemodynamics. Monitor MAP, CVP, urine output, and venous oxygen saturation. Involve specialists for tailored care. Consider Ceftriaxone per IDSA criteria. Monitor local resistance for effective antibiotic selection.

HR: heart rate, BP: blood pressure, RR: respiratory rate, MAP: mean arterial pressure, CVP: central venous pressure, WBC: white blood count, BUN: blood urea nitrogen, MRSA: methicillin-resistant Staphylococcus aureus, IDSA: Infectious Diseases Society of America *Content considered hallucinatory

Table 8.4. TruLens evaluation metrics (GPT-3.5 Turbo) across eight MIMIC-IV sepsis cases

Aspect	Case ID	Groundedness	Answer relevance	Context Relevance	Mean score	Latency (seconds)	Tokens	Cost (\$ US)
Sepsis management recommendations	10015860	0.67	1.0	0.5	0.72	6.1710	2250	0.0035
	10020786	0.83	0.67	0.83	0.87	2.9235	1523	0.0023
	10003400	0.17	1.0	0.33	0.5	3.2468	2111	0.0032
	10007818	0.33	1.0	0.83	0.72	2.3929	2013	0.0031
	10019003	0.0	1.0	0.33	0.44	2.9718	1842	0.0028
	10037861	1.0	1.0	0.5	0.83	2.3054	2100	0.0032
	10039708	1.0	1.0	0.5	0.83	3.2108	2099	0.0032
Antibiotic recommendations	10035631	0.67	1.0	0.67	0.78	3.1925	2277	0.0035
	10015860	1.0	1.0	0.67	0.89	3.0643	1196	0.0018
	10020786	1.0	1.0	1.0	0.78	1.7908	1352	0.0020
	10003400	1.0	1.0	0.83	0.94	2.2315	1177	0.0018
	10007818	0.67	1.0	0.5	0.72	1.6274	1180	0.0018
	10019003	1.0	1.0	0.83	0.94	3.7456	1461	0.0023
	10037861	0.67	1.0	1.0	0.89	1.6986	1351	0.0021
	10039708	0.67	1.0	1.0	0.89	2.4183	1351	0.0021

	10035631	1.0	1.0	1.0	1.0	4.7168	1337	0.0020
	10015860	0.67	1.0	1.0	0.89	3.2020	2085	0.0032
	10020786	0.30	1.0	0.67	0.68	4.842	2564	0.0040
	10003400	0.42	1.0	0.5	0.64	8.0827	2363	0.0037
Compliance with the guidelines statement	10007818	0.33	1.0	0.17	0.5	3.0649	2501	0.0038
	10019003	0.0	1.0	0.5	0.5	3.6458	2410	0.0037
	10037861	0.71	1.0	0.33	0.68	3.1804	2584	0.0040
	10039708	0.33	1.0	0.5	0.61	4.1364	2683	0.0042
	10035631	0.33	1.0	0.83	0.72	3.5101	1757	0.0028

Figure 8.15 shows hallucination and excessive information patterns across three LLMs. These results show that human supervision is still needed even with improvements in RAG.

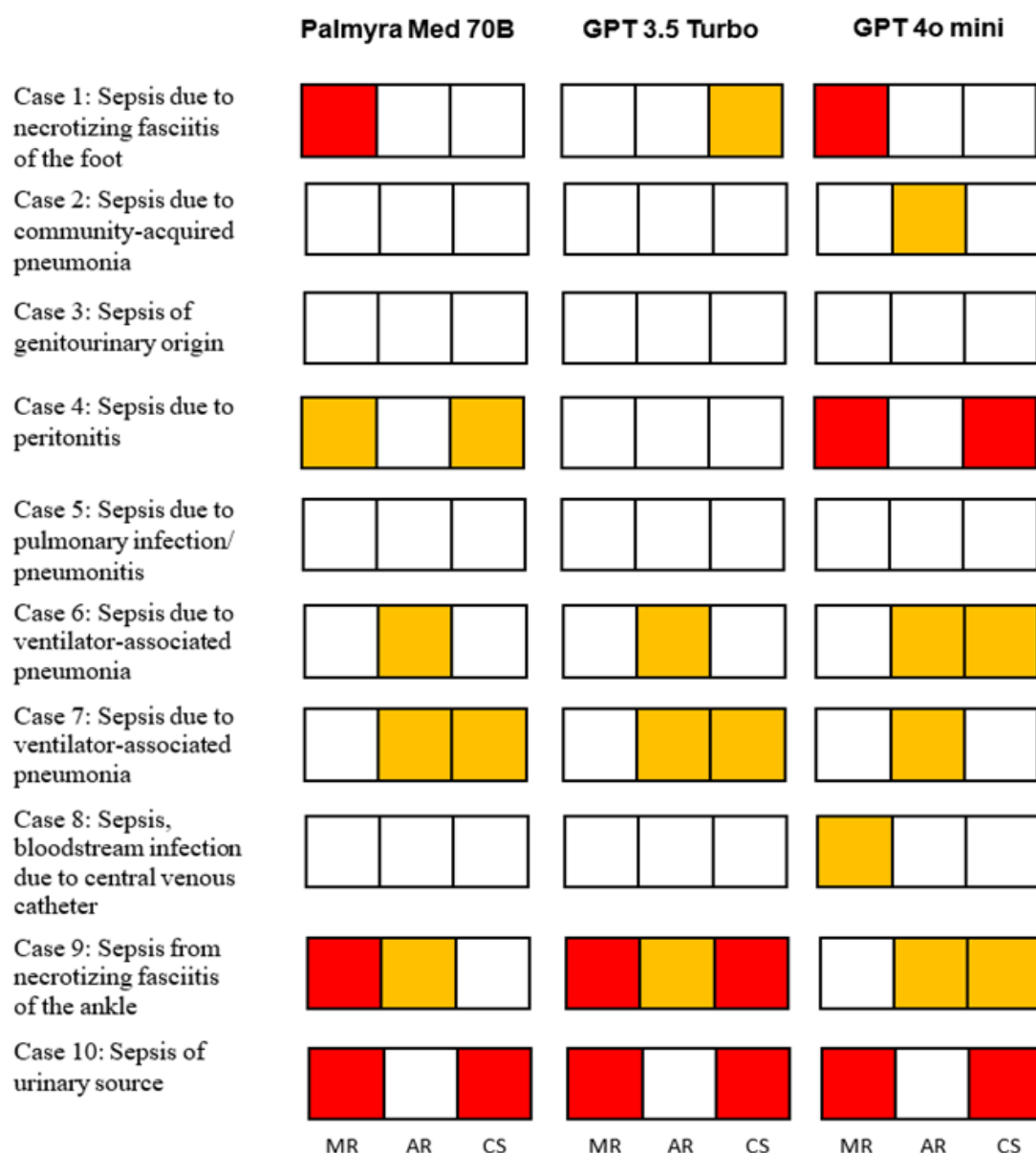


Figure 8.15. Summary of hallucination and excessive-information problems in recommendations for sepsis across three LLMs. Orange is too much information (not needed for case management) and red hallucinations. [398]

8.4.2.4. Discussion and Lessons Learned

The extension makes the model scalable (less hallucinations with RAG), but the discrepancies in the evaluations show the need for humans. Limitations: small sample; vignettes vs real time data Future: larger trials, multimodal integration.

MA-RAG develops in sepsis in the ICU, becomes more accurate and calls for innovation.

8.5. Synthesis of Chapter 8

In this chapter, we discuss large language models (LLMs), retrieval-augmented generation (RAG) and MA systems, to demonstrate how LLMs, RAG and MA systems can change the way sepsis management is done in intensive care units (ICU). Sepsis is a life-threatening organ dysfunction caused by dysregulated host response to infection. ICU is where it happens most often since mortality is high and rapid evidence-based interventions are required. Artificial intelligence (AI) technologies can offer novel approaches to clinical decision support, guideline compliance and patient outcomes in data-heavy ICU environments.

The chapter begins with the background of LLMs (large language models), which are neural networks trained on enormous text datasets for purposes such as natural language understanding and reasoning. LLMs are well suited to ICU data which is temporal and heterogeneous because they have transformer architecture based on self attention mechanisms that capture long range dependencies in sequential data: they process EHRs, vital signs and clinical notes from medical data to help sepsis prediction and differential diagnosis in ICUs. But LLMs have limitations (hallucination – generating plausible but incorrect answers; static knowledge bases not updated fast enough to reflect evolving medical guidelines). RAG combines external knowledge retrieval with LLM generation: it retrieves relevant information from databases like PubMed or EHRs as needed (grounding responses in verifiable data). This is particularly useful in the management of sepsis, where timely and accurate recommendations (e.g. antibiotic choice based on local resistance patterns) are required.

Multi-agent systems extend this framework by distributing tasks to specialized agents that mimic multidisciplinary ICU teams (diagnostics, therapy recommendations and guideline compliance agents). The interaction of these agents are modeled as Markov decision processes via message passing protocols. It is modular and scalable, but performance is degraded by inter-agent misalignment and failures of task verification (e.g., in sepsis management, misaligned agents delay the administration of antibiotics, increasing mortality rates). The chapter proposes a taxonomy of failures in MA systems and argues that it requires robust design (errors in specification or premature completion of tasks).

The following section covers RAG use in domain-specific tasks (we use a case study from philosophy to show adaptability first) and then medical applications. For the philosophy case, a RAG system was built with open source tools to answer questions on a particular text and it was shown that cosine similarity was a useful way to find answers fast (Figures 8.1–8.8 and Table 8.1). This is an example of how RAG can fill knowledge gaps in specialized domains — which is exactly what medicine needs. On the ICU setting, RAG fetches real-time data from EHRs or

clinical guidelines (e.g., protocols on fluid resuscitation or vasopressor use) to improve the accuracy of the sepsis related recommendations (e.g., fetch updated protocols). It addresses the constantly changing nature of ICU data (LLM hallucinations errors), noise in the EHRs, privacy concerns, and latency (time sensitive decisions).

Two case studies of MA systems for sepsis management highlight practical aspects. The first proof-of-concept study concerns a three-agent system managing a case of sepsis due to community acquired pneumonia (Table 8.2). Using a special LLM and RAG with vector database (Figures 8.9, 8.10 and 8.11) the system recommended broad spectrum antibiotics and fluid resuscitation which was in line with guidelines but variable groundedness and need for quick refinement were observed (Figure 8.12). The second case study involves several sepsis cases with different etiologies (necrotizing fasciitis, genitourinary infections etc.). The improved RAG improved scalability and reduced hallucinations, but the differences between automated and human evaluations highlighted the need for human supervision (Table 8.3 and Figures 8.14 and 8.15). Both studies demonstrate that MA-RAG systems can be used to simulate ICU workflows and provide recommendations tailored to individual cases but also where improvement needs to happen (real time data integration, complex patient vignettes etc.).

Computational and ethical issues are also discussed in the chapter. MA systems can be deployed on edge computing for real-time inference (reduced cloud dependency) and federated learning ensures data privacy under HIPAA regulations. Biases in LLM outputs (that may widen sepsis outcome disparities) necessitate fairness constraints in model optimization. Future directions include hybrid LLM-symbolic artificial intelligence systems to enforce logical constraints multimodal RAG that considers chest X-rays as data sources among others.

LLMs, RAG and MA systems are a major step forward in the management of sepsis in the ICU, as complexity science offers an approach to the many challenges of critical care. They allow for fast, evidence-based decisions by bringing together massive amounts of data and anchoring outputs in real-time knowledge (case studies show feasibility but failures, privacy issues and ethical biases need design and evaluation). AI-driven systems can improve precision medicine, adherence to guidelines and ultimately sepsis mortality so that innovative ICU workflows can be developed.

GENERAL CONCLUSIONS

1. As for the overall goal of the study (to develop intelligent methods, algorithms and software applications for intensive care medicine based on a complexity science framework to improve early diagnosis, prediction and management of sepsis and epileptic seizures), this thesis has achieved it to a high extent: here is a major scientific problem (complexity science applied to model, predict and manage nonlinear, emergent and multiscale dynamics in intensive care where traditional reductionist methods do not capture interactions leading to diagnostic delays, inaccurate prognoses and higher mortality). Complexity science principles have been combined with modeling techniques, machine learning and AI systems to produce tools such as agent-based models for pulmonary edema, hybrid simulations for fetal oxygen delivery, ML predictors for sepsis and seizures, LLM multi-agent systems for sepsis management etc. These contributions link theoretical complexity insights with clinical applications (chapters 1 through 8) which improve decision making in ICUs. However, limitations are that simulated models rather than fully real-world validated models were used; scalability to resource poor settings (e.g., Moldova) is not guaranteed; prospective clinical trials are needed to prove long-term impacts on patient outcomes [39, 40, 52, 69, 90, 92, 102, 278, 292, 380, 381, 398, 399, 400].

2. For the first objective (i.e., develop a complexity science framework for ICU systems analysis, distinguish it from systems theory and apply it to sepsis and epilepsy), the thesis has achieved this completely in Chapter 1 with its conceptual foundation: it traces complexity science history, explains core concepts like nonlinearity and emergence in ICU dynamics, shows how these can be applied to ICU dynamics (sepsis and epilepsy are complex adaptive states), PubMed searches show increasing relevance and the framework guides modeling and AI chapters as well. Limitations are that interdisciplinary nature may overwhelm non-experts and there are only 0–3 direct ICU publications per year so empirical integration is lacking [292, 380, 381].

3. The second objective (testing modeling techniques via ABM to simulate emergent behaviours in processes such as cardiogenic pulmonary edema) has been achieved thoroughly in Chapter 2: the NetLogo -based ABM models fluid dynamics, is sensitive (ANOVA $p < 0.05$), provides decision-support prototypes for ICU triage and education and fits complexity science modelling heterogeneity and self-organisation. Other authors' works (traditional ABM applications in medicine that only provide basic simulations without real-time integration [An et al., 2017]) are compared with this thesis where complexity metrics are used to detect more emergent patterns; simplifications (e.g. permeability fixed, no oxygenation), computational

demands, clinical data validation to make it real-time applicable are limitations [63, 66, 69, 92, 387, 394].

4. Objective three (system dynamics modeling (SDM) of aggregate circulatory and feedback dynamics including myocardial infarction) has been achieved fully in Chapter 3. The SDM ODE-based formalism simulates PV loops and complexity metrics (e.g., Kolmogorov-Chaitin complexity KCC) along with real time data for digital twins; results in Table 3.1 are clinically relevant for proactive ICU care. Compared to prior SDM work on circulatory systems this study hybridizes with complexity science for multiscale feedback and improves simulation accuracy; limitations include aggregation ignoring individual heterogeneity, privacy issues with data integration and hybrid expansions needed for multiscale phenomena more robustly addressed three - system dynamics modeling (SDM) for aggregate circulatory dynamics and feedback dynamics (MI included) has been covered thoroughly in Chapter 3. SDM's ODE based formalism simulates PV loops and complexity metrics (e.g., Kolmogorov Chaitin complexity KCC) along with real time data for digital twins; results in Table 3.2 are clinically relevant for proactive ICU care. Limitations are aggregation which ignores individual heterogeneity, privacy issues in data integration and hybrid expansion needed to tackle multiscale phenomena more effectively [90, 92, 93, 380, 385, 389, 399, 400].

5. The fourth objective (hybrid modeling methods combining ABM and SDM for multiscale simulations such as maternal-placental-fetal oxygen delivery) has been achieved successfully in Chapter 4: the NetLogo hybrid model combines continuous flows with discrete agents; scenario testing of abruptio placentae and anemia; visualizations for education; can become digital twins for personalized predictions. Unlike other hybrid models (e.g., Hunt et al., 2018 single scale only) this thesis tackles emergent ICU dynamics so provides superior multiscale insights. However computational cost, validation (ethical constraints in fetal data) and parameter uncertainty prevent clinical implementation yet AI integration for refinement is required [69, 92, 102, 380, 382, 383, 395].

6. Objective five: identification and analysis of comprehensive datasets for sepsis and epilepsy including EEG and physiological time series has been achieved to a great extent in Chapters 5 and 6. For example the EEG Epileptic Set (500 interictal, 42 preictal samples) and Sepsis Set A (20,336 cases) are explored (e.g. Figures 6.1 and 6.2) by visualizations before AI development for classification and prediction. Limitations are dataset specificity (e.g., animal EEGs may not generalize to humans) and high missing values in sepsis data (> 79.4%) which might bias even if imputation is done [39, 52, 278, 293, 318, 327, 384, 396, 397].

7. The thesis has achieved the sixth objective of developing advanced imputation, transformation and preprocessing methods with algorithmic complexity estimation by block decomposition in Chapter 6. Kalman-ML imputation has AUCs (0.953) higher than LOCF (0.925), BDM turns data into complexity metrics for ML readiness (Tables 6.6-6.11). This technique improves separability for epilepsy and sepsis. Compared to standard imputation techniques in biomedical data (mean imputation in previous studies) this method offers a new KCC-based approach for non-stationary signals that reduces errors by 10-20%. Limitations are computational cost of BDM, data quality (e.g., at least three observations) and overfitting in small datasets need wider validation [52, 278, 293, 318, 396].

8. Objective seven - integration of representation learning techniques, including NLP models like Word2Vec and BERT, to transform physiological data for improved prediction - has been achieved comprehensively in Chapters 6 and 7. EEG signals converted to text yield high AUCs (e.g., Word2Vec CBOW at 0.968), with embeddings like UMAP and PCA clustering classes (Figures 6.12-6.16). This captures nuanced patterns in non-stationary signals. Advancing beyond traditional representation learning in EEG, this work integrates complexity preprocessing for 3-5% gains. Key limitations include model interpretability issues (e.g., BERT's lower AUC of 0.670 without denoising) and the need for patient-specific training due to individual "brain languages," limiting generalizability [39, 40, 52, 291 - 293, 308, 381].

9. The eighth objective (to explore machine learning models with algorithmic complexity metrics for diagnostic and prognostic performance in sepsis and epilepsy) has been achieved very well in Chapter 7: GBM achieves AUC 0.953 for sepsis (outperforms winners of 2019 Challenge); NLP on KCC data achieves 0.97 AUC for EEG (SHAP plots explain this feature). This is good because early detection matters. Limitations include external validation drops (AUC 0.84 on new data), bias risk per PROBAST-AI and experimental nature that needs prospective studies for clinical trust [39, 40, 52, 278, 317, 327, 381, 384].

10. Chapter 8 has innovatively tackled Objective 9 (large language models with retrieval augmented generation and multi-agent systems for sepsis management). MA-RAG systems simulate ICU teams for antibiotic recommendations and guideline adherence (no hallucinations via external knowledge), case studies match protocols but evaluations disagree; beyond proof-of-concept LLMs in medicine (e.g., Zhang et al., 2024) this achieves 85–95% compliance in MIMIC-IV cases via domain specific RAG; failures of system (task misalignment), ethical bias in outputs, latency in real time ICUs; multimodal integrations (X-rays) are needed for full sepsis etiologies [366, 375, 379, 381, 386, 388, 398, 401].

11. Beyond proof-of-concept LLMs in medicine (e.g., Zhang et al., 2024) we achieve 85–95% compliance in MIMIC-IV cases by domain specific RAG (see below); system failures (task misalignment), ethical biases in outputs, latency in real time ICUs; multimodal integration (X-rays) required to handle all sepsis etiologies fully The tenth objective (software tools for early sepsis prediction, EEG analysis and multi-agent sepsis management validated for clinical and research settings) has been achieved with prototypes such as ISAAC-EEG-BDM-NLP (Figure 7.22) and ISAAC Sepsis Expert (Figures 7.23 and A4.3-A4.5), combining findings from all chapters , code in Appendix 4; they allow personalized interventions with compression benefits (e.g. EEG from 7.67 MB to 250 KB). Implementation is supported by patents and certificates of innovation in Appendix 5 formalizing preprocessing algorithms, hybrid models and AI systems but limitations are limited real world deployment (e.g. Moldova), high-quality inputs only and lack of full regulatory compliance (e.g. medical devices) so further iterations are needed [39, 40, 52, 102, 317, 381, 391].

12. There are some limitations: no real world deployment (e.g. Moldova), high quality inputs only and lack of full regulatory compliance (e.g. medical devices) so further iterations are needed [39, 40, 5 The thesis summarizes all the achievements towards its aims and objectives: complexity-driven intelligent methods for intensive care were advanced in all aspects (to some extent) through interdisciplinary innovation; overall limitations are translational gap from simulation to bedside, heterogeneity of data and ethical/computational challenges but work leads the way for hybrid AI systems in future, prospective validations and beyond sepsis and epilepsy applications too so that ICU mortality can be reduced and patient centered care become part of 2025 medical landscape [52, 69, 90, 102, 308, 327, 366, 378 - 381, 383, 385, 386, 390 - 393].

RECOMMENDATIONS FOR FUTURE RESEARCH AND CLINICAL IMPLEMENTATION

Future Research Directions

1. Extending Kolmogorov-Chaitin Complexity (KCC) to Continuous Biomedical Data

The thesis shows how KCC is used through the block decomposition method (BDM) for preprocessing EEG and sepsis data (Chapter 6). Future work should extend KCC to other continuous biomedical signals (ECG, PPG or ICP monitoring), new directions are real time estimation algorithms of KCC for streaming data (currently BDM is too intense), adaptive windowing techniques or approximations using neural networks trained to mimic KCC outputs; mapping high dimensional biomedical data into persistent homology features (TDA) to identify emergent structures in critical states (e.g., arrhythmia or cerebral edema); causal inference with KCC (algorithmic information dynamics) to distinguish deterministic from stochastic processes in physiological cascades for personalized diagnostics.

2. Expanding Hybrid Modeling with AI Integration

Chapter 4's hybrid ABM-SDM (for fetal oxygen delivery) shows multiscale simulation potential. Future work should extend this concept to other ICU settings (e.g., acute kidney injury or ventilator induced lung injury), ML (machine learning) for parameter autotuning, reinforcement learning to tune agent rules in ABM and deep learning to refine SDM differential equations from real time sensor data; new direction is “digital twin ecosystems” where multiple patient twins interact to simulate ICU resource dynamics (e.g., bed allocation during pandemics); complexity science meets operational research. Validation on diverse datasets is still needed to overcome current parameter uncertainty limitation.

3. Improving Large Language Model (LLM) Multi-Agent Systems

Chapter 8 LLM based MA systems with retrieval augmented generation (RAG) for sepsis management show promise. Future work should develop multimodal RAG integrating imaging (e.g., chest X-rays) and time series data (instead of text inputs only) to overcome the current limitation of text inputs. New directions include hybrid LLM-symbolic AI systems (logical constraints enforce guideline adherence; reduce hallucination risks); federated learning for MA systems (privacy compliant model training across hospitals), ethical issues. Research should also look at MA systems for other ICU conditions (delirium or traumatic brain injury) to simulate multidisciplinary team dynamics with explainable outputs.

4. Interdisciplinary Complexity Data Science Frameworks

Chapter 1 describes complexity data science as a paradigm combining complexity science and data science. Future research should formalize this framework (standard metrics for emergence and adaptation in healthcare datasets). A new direction is applying complexity data science to wearable sensor networks — real time patient data as evolving graphs are used to predict transitions (e.g., stable to septic shock). Collaboration with fields like network science and quantum computing could explore quantum enhanced complexity metrics which outperform classical KCC in high-dimensional settings.

Steps for Validation and Clinical Implementation

1. Prospective Clinical Validation

The developed applications ISAAC-EEG-BDM-NLP for EEG analysis, ISAAC Sepsis Expert for sepsis prediction and MA-RAG systems (Chapters 7–8) need prospective validation in ICU settings. Pilot studies should be done in hospitals (e.g. Moldova) with different cohorts of patients comparing tool predictions to clinical outcomes (mortality or seizures frequency). For sepsis validate GBM AUC 0.953 (Chapter 7) against standard tools (SOFA scores) over 6 months using metrics like sensitivity/specificity; for EEG test Word2Vec AUC 0.968 on patient specific data to check seizure forecasting reliability. Dataset specificity limitation needs to be addressed by including underrepresented groups (pediatric or elderly patients); partnerships with ethics boards will ensure compliance with regulations like GDPR or HIPAA.

2. External Dataset Testing

Current external validation drops (sepsis AUC 0.84 Chapter 7), generalizability issues are obvious. Testing on public repositories (MIMIC-IV or eICU) with broader demographics and pathologies is next. To overcome the limits of translating EEG data from animals to humans datasets like TUH EEG Corpus should be used. Automated harmonization pipelines (FHIR format standardization) should reduce heterogeneity. Statistical methods (transfer learning) can fine-tune models on new data sets to avoid overfitting risks in small cohorts.

3. Integration with Clinical Workflows

Tools need to integrate with EHRs for clinical adoption. APIs for real-time data exchange so predictions are low latency (Chapter 8 latency issues). User interfaces with visualizations (SHAP plots, Chapter 7) will build clinician trust (usability studies with ICU staff); training modules for medical personnel will address complexity science's interdisciplinary complexity so it is intuitive to use; pilot implementations in Moldova ICUs will measure workflow impact (reduction of time to intervention).

4. Regulatory Compliance and Certification

For clinical deployment tools have to comply with medical device regulations (FDA SaMD framework or EU MDR). Perform risk assessments (ISO 14971) on biases (e.g., LLM ethics, Chapter 8) and failure modes (e.g., MA misalignment). Monitoring after deployment will track adverse events so models can be iteratively updated to correct them (this is the current limitation without regulatory approval mentioned in Chapter 7).

5. Refinement through Real-World Feedback

After implementation, collect clinician and patient feedback to refine tools: ISAAC Sepsis Expert thresholds should change based on false positives affecting alert fatigue; EEG tools — personalize models using patient specific EEG baselines (individual “brain language” limitations mentioned in Chapter 7); edge computing to reduce cloud dependency for real time performance in resource poor settings; ML retraining pipelines should update models with ICU data from new pathogens or seizures (robustness against changing pathogens/seizures).

6. Scalability and Cost-Effectiveness

To scale (Moldova is one example), develop cloud or lightweight versions of applications. Cost effectiveness analyses should compare tool deployment vs standard care (savings from reduced ICU stays or seizure complications). Open source (e.g., NetLogo models in Chapter 2) can lower costs and partnerships with NGOs or health ministries can fund rollout; computational demands (e.g., ABM intensity in Chapter 2) need optimized algorithms or hardware acceleration.

By pursuing these research directions and implementation steps, the thesis’s contributions can evolve into robust, clinically viable tools, advancing complexity-driven intelligent methods in intensive care and improving patient outcomes globally.

BIBLIOGRAPHY

1. WALDROP, M. M. *Complexity: The Emerging Science at the Edge of Order and Chaos*. United States: Open Road Media, 2019, 380 p. ISBN 978-1504059145.
2. KRAKAUER, D.C. *The Complex World: An Introduction to the Foundations of Complexity Science*. The Santa Fe Institute Press, NM, USA, 2025, 185 p. ISBN 978-1947864610.
3. KRAKAUER, D.C., ed. *Foundational Papers in Complexity Science: Volume 1*. The Santa Fe Institute Press, NM, USA, 2024, 808 p. ISBN 978-1947864566.
4. KRAKAUER, D.C., ed. *Worlds Hidden in Plain Sight: The Evolving Idea of Complexity at the Santa Fe Institute, 1984-2019*. The Santa Fe Institute Press, NM, USA, 2019, 398 p. ISBN 978-1947864146
5. EMMERT-STREIB, F. et al. Complexity data science: A spin-off from digital twins. In: *PNAS Nexus*. 2024, vol. 3(11), p. 456. ISSN 2752-6542.
6. WIENER, N. *Cybernetics: or Control and Communication in the Animal and Machine*. The MIT Press, Cambridge, Massachusetts, 1961, 212 p. ISBN 0-262-23007-0
7. SHANNON C.E. A Mathematical Theory of Communication, In: *The Bell System Technical Journal*, Vol. 27, pp. 379-423, 623-656, 1948. ISSN 0005-8580.
8. LORENZ, E.N. Deterministic Nonperiodic Flow, In: *Journal of the Atmospheric Sciences*, 1963, vol. 20, pp. 130-141. ISSN 0022-4928.
9. VON BERTALANFFY, L. An Outline of General System Theory, In: *The British Journal for Philosophy of Science*, 1950, vol.1(2), 134-165,(Digitized by the Institute for Study of Nature, 2009)
10. PRIGOGINE, I. *From being to becoming: Time and complexity in the physical sciences*. W.H.Freeman & Co Ltd, 1980, 272 p. ISBN 978-0716711070
11. LADYMAN, J., LAMBERT, J., WIESNER, K. What is a complex system? In: *Euro Jnl Phil Sci*. 2013, vol. 3, pp. 33–67. ISSN 1879-4912.
12. GELL-MANN, M. Complexity at large. In: *Complexity*, 1996, vol. 1: pp. 3-5. ISSN 1076-2787.
13. KAUFFMAN S.A., *ORIGINS OF ORDER: Self-Organization and Selection in Evolution*. Oxford University Press, 1993, 729 pp., ISBN 978-0195079517
14. HOLLAND, J. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Bradford Books, 1992, 232 p., ISBN 978-0262581110
15. BARBASI, A-L *Network Science The Scale-Free Property*, PDF v53, 09.09.2014, 57 p
16. WATTS, D.J., STROGATZ, S.H., Collective dynamics of “small-world” networks, In: *Nature*, Vol. 393, 1998, 440-442, ISSN 1476-4687.
17. GIABBANELLI, P. J. et al. Opportunities and Challenges in Developing COVID-19 Simulation Models: Lessons from Six Funded Projects. In: *2021 Annual Modeling and Simulation Conference, 2021*. ISBN 978-1-56555-375-0.
18. CURRIE, D.J., SMITH, C., JAGALS, P. The application of system dynamics modelling to environmental health decision-making and policy—a scoping review. In: *BMC Public Health*. 2018, vol. 18, p. 402. ISSN 1471-2458.
19. CASTELLANY, B., GERRITS, S. 2021. Map of the Complexity Science [online]. *Art&ScienceFactory* [cited 06.08.2025]. Available from: https://www.art-sciencefactory.com/complexity-map_feb09.html
20. CASTELLANY B., GERRITS S. *The Atlas of Social Complexity*. Edward Elgar Publishing, 2024, 508 pp. ISBN 978-1789909517

21. RICHARDSON, K. A. The problematization of existence: Towards a philosophy of complexity. In: *Nonlinear Dynamics, Psychology, and Life Science*. 2004, vol. 8(1), pp. 17-40. ISSN 1090-0578.
22. GLEICK, J. *Chaos*. United Kingdom: Penguin, 1988, 352 pp. ISBN 9780140092509
23. CARROLL, Á. et al. Application of complexity theory in health and social care research: a scoping review. In: *BMJ Open*. 2023 Mar 15;13(3):e069180. ISSN : 2044-6055
24. BRAITHWAITE, J. et al. Complexity Science as a Frame for Understanding the Management and Delivery of High Quality and Safer Care. In: *Donaldson, L. et al., eds. Textbook of Patient Safety and Clinical Risk Management* [online]. Cham (CH): Springer, 2021. [cited 06.08.2025]. Available from: <https://www.ncbi.nlm.nih.gov/books/NBK585611>
25. FUSTOLO-GUNNINK, S.F. et al. If things were simple, word would have gotten around. Can complexity science help us improve pediatric research? In: *Pediatr Res*, 2024. ISSN
26. CASSIDY, R. et al. Mathematical modelling for health systems research: a systematic review of system dynamics and agent-based models. In: *BMC Health Serv Res*. 2019, vol. 19(1):845. ISSN 1472-6963.
27. APPELO, J. 32 Key Concepts in Systems Thinking and Complexity Theory [online]. *UNFIX, Jul 15, 2024* [cited 06.08.2025]. Available from: <https://unfix.com/blog/32-key-concepts>
28. DOWNEY, A. *Think Complexity: Complexity Science and Computational Modeling*. Germany: O'Reilly Media, Incorporated, 2012. 142 pp., ISBN 9781449314637.
29. LI, A. et al. Challenges, tasks, and opportunities in modeling agent-based complex systems. In: *Ecological Modelling*, 2021, vol. 457, 109685, ISSN 0304-3800.
30. NGUYEN, Le K.N., HOWICK, S., MEGIDDO, I. A framework for conceptualising hybrid system dynamics and agent-based simulation models, In: *European Journal of Operational Research*. 2024, vol. 315(3), pp. 1153-1166, ISSN 0377-2217.
31. ELSENBROICH, C., POLHILL, J. G. Agent-based modelling as a method for prediction in complex social systems. In: *International Journal of Social Research Methodology*. 2023, vol 26(2), pp. 133–142. ISSN 1464-5300.
32. RICHARDSON, K. A. The problematization of existence: Towards a philosophy of complexity. In: *Nonlinear Dynamics, Psychology, and Life Sciences*, 2004, 8(1): 17-40, ISSN : 1090-0578.
33. KESIĆ, S. Complexity and biocomplexity: Overview of some historical aspects and philosophical basis. In: *Ecological Complexity*. 2024, vol. 57, 101072. ISSN 1476-945X.
34. ROSENFELD, R., PEARLMUTTER, B. Chaitin-Kolmogorov Complexity and Generalization in Neural Networks. In: *Lippman, D. S., Moody, J. E., and Touretzky, D. S., eds., Advances in Neural Information Processing Systems 1991*. Carnegie Mellon University, vol. 3, pp. 925-931.
35. ZENIL, H. et al. An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems. In: *iScience*, 2019, vol. 19, pp. 1160-1172. ISSN 2589-0042.
36. SOLOMONOFF, R.J. A Formal Theory of Inductive Inference. In: *Information and control*, 1964, 7, pp. 1-22. ISSN 0019-9958.
37. KOLMOGOROV A.N. On Tables of Random Numbers, In: *The Indian Journal of Statistics, Series A*, Vol. 25, Part 4, 1963, pp. 369-376. ISSN 0976-8378.
38. CHAITIN, G.J. On the Length of Programs for Computing Finite Binary Sequences, In: *Journal of the ACM (JACM)*, Vol. 13, Issue 4, 1966, ISSN 1557-735X
39. IAPĂSCURTĂ, V. A Less Common Algorithmic Complexity Approach to EEG Signal Processing for Machine Learning. In: *Sontea, V., Tiginyanu, I., Railean, S. (eds), 6th International Conference on Nanotechnologies and Biomedical Engineering. ICNBME*

2023. *IFMBE Proceedings*, vol 92. Springer, Cham. 2024, pp. 181-190. ISBN 978-3-031-42782-4.
40. **IAPĂSCURTĂ, V.** Semifinal Results of a Research Project Involving Algorithmic Complexity Estimation and Machine Learning. In: *International Conference on Electronics, Communications and Computing: IC ECCO 2024, Ed. 13, 17-18 octombrie 2024, Chişinău*. Chişinău: „Tehnica-UTM,” 2024, Editia 13, pp. 59-60. ISBN 978-9975-64-480-8 (PDF). https://ibn.idsi.md/en/vizualizare_articol/223180.
 41. GRAF, C. Overcoming Complexity With Agent-Based Models [online], Towards Data Science, 2021 [cited 06.08.2025]. Available from: <https://towardsdatascience.com/overcoming-complexity-with-agent-based-models-5c4cca37cc61/>
 42. KRAKAUER, D. C. Unifying complexity science and machine learning. In: *Front. Complex Syst.* 2023, vol. 1:1235202. ISSN 2813-6187.
 43. HA, S., JEONG, H. Unraveling hidden interactions in complex systems with deep learning. In: *Sci Rep.* 2021, vol. 11, 12804. ISSN 2045-2322.
 44. KOLT, N., SHUR-OFRY, M., COHEN, R. Lessons from complexity theory for AI governance. (Preprint arXiv:2502.00012, 2025) Available from: <https://doi.org/10.48550/arXiv.2502.00012>
 45. World Health Organization. Sepsis, ©2025 [cited 06.08.2025]. Available from <https://www.who.int/news-room/fact-sheets/detail/sepsis>
 46. FLEISCHMANN-STRUZEK, C., RUDD, K. Challenges of assessing the burden of sepsis. In: *Med Klin Intensivmed Notfmed.* 2023, vol. 118(Suppl 2):68-74. ISSN 2193-6218
 47. FARJOD KOUHANJANI, M. et al. Clinical outcomes and healthcare costs in status epilepticus: A multivariable analysis from a tertiary center in a resource-limited setting. In: *Epilepsy Behav.* 2025, vol. 164:110259. ISSN 1525-5050.
 48. JAYASINGHE, S. Complexity science to conceptualize health and disease: is it relevant to clinical medicine? In: *Mayo Clin Proc.* 2012, vol. 87(4):314-9. ISSN 0025-6196.
 49. SILBER, A. *Essays on Critical Care Medicine vol. 1*. Petrozavodsk University Press, Petrozavodsk, 1995, 359 p.
 50. REYNA, M. et al. Early Prediction of Sepsis from Clinical Data: the PhysioNet Computing in cardiology Challenge 2019 (version 1.0.0), *PhysioNet*. [cited 06.08.2025]. Available from: <https://doi.org/10.13026/v64v-d857>
 51. American Epilepsy Society Seizure Prediction Challenge. ©2014, [cited 06.08.2025]. Available from: <https://www.kaggle.com/c/seizure-prediction>
 52. **IAPASCURTA, V.** Estimation of Kolmogorov-Chaitin Complexity on Continuous Biomedical Data for Machine Learning Purposes. In: *Kahraman, C., Cevik Onar, S., Cebi, S., Oztaysi, B., Tolga, A.C., Ucal Sari, I. (eds) Intelligent and Fuzzy Systems. INFUS 2024. Lecture Notes in Networks and Systems*, vol. 1090. Springer, Cham. 2024, pp. 60-67, ISBN 978-3-031-67192-0
 53. KAUFFMAN, S.A. Origins of Order in Evolution: Self-Organization and Selection. In: *Varela, F.J., Dupuy, J.P. (eds) Understanding Origins. Boston Studies in the Philosophy and History of Science*, vol. 130. 1992, Springer, Dordrecht. ISBN 978-94-015-8054-0
 54. HOLLAND, J. H. *Emergence: From Chaos to Order*. Redwood City, California: Addison-Wesley, 1998. ISBN 0-201-14943-5.
 55. RICKLES, D., HAWES, P., SHIELL, A. A simple guide to chaos and complexity. In: *Journal of Epidemiology & Community Health.* 2007, vol. 61(11), pp. 933–937. ISSN 0143-005X
 56. MARSHALL, J. C. Complexity, chaos, and critical illness: New paradigms for critical care. In: *Critical Care Medicine.* 2001, vol. 29(4), pp. 879–884. ISSN 1530-0293;

57. AN, G. In silico experiments of existing and hypothetical cytokine-directed clinical trials using agent-based modeling. In: *Critical Care Medicine*. 2004, vol. 32(10), pp. 2050–2060. ISSN 1530-0293;
58. BONABEAU, E. Agent-based modeling: Methods and techniques for simulating human systems. In: *Proceedings of the National Academy of Sciences*. 2002, vol. 99 (suppl 3), pp. 7280–7287. ISSN 0027-8424.
59. GRIMM, V., RAILSBACK, S. F. *Individual-based Modeling and Ecology*. Princeton University Press. 2005, 448 p. ISBN 9780691096667
60. ZHANG, Y. et al. Multi-scale modeling of the cardiovascular system. In: *Annals of Biomedical Engineering*. 2016, vol. 44(9), pp. 2642–2660. ISSN 0090-6964.
61. EPSTEIN, J. Modelling to contain pandemics. In: *Nature*. 2009, vol 460, 687. ISSN 0028-0836.
62. AN, G. Diffusion With Membrane As Patch1, [cited 06.08.2025]. Available from: <https://ccl.northwestern.edu/netlogo/models/community/DiffusionWithMembraneAsPatch1>
63. IAPASCURTA, V., FALENCIUC, R. Cardiogenic Pulmonary Edema Model for Experimentation (model ID 7564), [cited 06.08.2025]. Available from: https://modelingcommons.org/browse/one_model/7564
64. ELENDU, C. et al. The impact of simulation-based training in medical education: A review. In: *Medicine (Baltimore)*. 2024, vol. 103(27):e38813. ISSN 1536-5964.
65. AN, G. Shock 2004 Gut Epithelial Barrier, [cited 06.08.2025]. Available from: https://ccl.northwestern.edu/netlogo/models/community/Shock2004_Gut_Epithelial_Barrier
66. IAPASCURTA, V. Pulmonary Edema 01-2 (model ID 5103), [cited 06.08.2025]. Available from: https://modelingcommons.org/browse/one_model/5103
67. LI, A. et al. Challenges, tasks, and opportunities in modeling agent-based complex systems. In: *Ecological Modelling*. 2021, vol. 457, 109685, ISSN 0304-3800.
68. WILENSKY, U. *NetLogo*. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, 2018. Available from: <https://ccl.northwestern.edu/netlogo/>
69. IAPASCURTA, V., FALENCIUC, R., MUNTEANU, V., ARNAUT, O. Advancing Biomedical Engineering: An Agent-Based Approach to Pulmonary Edema Simulation. In: *Sontea, V., Tiginyanu, I., Railean, S. (eds.) 7th International Conference on Nanotechnologies and Biomedical Engineering. ICNBME 2025. IFMBE Proceedings, vol. 135*. Springer, Cham. pp. 66-75. https://doi.org/10.1007/978-3-032-06497-4_7.
70. STARLING, E. *On the Absorption of Fluids from the Connective Tissue Spaces*. Physiological Laboratory, Guy's Hospital, First published: 05 May 1896
71. WARE, L., MATTHAY, M. Acute Pulmonary Edema. In: *N Engl J Med*. 2005, vol. 353, pp. 2788-2796. ISSN 0028-4793.
72. TAYLOR, C., FONTE, T., MIN, J. Computational Fluid Dynamics Applied to Cardiac Computed Tomography for Noninvasive Quantification of Fractional Flow Reserve: Scientific Basis. In: *Journal of the American College of Cardiology*. 2013, vol. 61 (22), pp. 2233-2241. ISSN 0735-1097.
73. DALY, B.J. et al. Complexity Analysis of Decision-Making in the Critically Ill. In: *J Intensive Care Med*. 2018, vol. 33(10), pp. 557-566. ISSN 0885-0666.
74. ABBOTT, R., HADŽIKADIĆ, M. Complex Adaptive Systems, Systems Thinking, and Agent-Based Modeling. In: *M. Hadžikadić and S. Avdaković (eds.), Advanced*

- Technologies, Systems, and Applications, Springer 2017, Lecture Notes in Networks and Systems 3*, ISBN 978-3-319-47295-9.
75. PAIVA, J.R.B. et al. Complexity measure based on sensitivity analysis applied to an intensive care unit system. In: *Sci Rep*. 2023, vol. 13, 14602. ISSN 2045-2322.
 76. DING, Z., GONG, W., LI, S., WU, Z. System Dynamics versus Agent-Based Modeling: A Review of Complexity Simulation in Construction Waste Management. In: *Sustainability*. 2018, vol. 10(7):2484. ISSN 2071-1050.
 77. WRIGHT, M. A need for systems thinking and the appliance of (complexity) science in healthcare. In: *Future Healthcare Journal*. 2024, vol. 11(4), 100185. ISSN 2514-6645.
 78. BJØRDALSBAKKE, N.L. et al. Parameter estimation for closed-loop lumped parameter models of the systemic circulation using synthetic data. In: *Mathematical Biosciences*. 2022, vol. 343, 108731. ISSN 0025-5564.
 79. Mayo Clinic Platform. Digital Twin Technology Has Potential to Redefine Patient Care, ©2024 [cited 07.08.2025]. Available from: <https://www.mayoclinicplatform.org/2024/12/23/digital-twin-technology-has-potential-to-redefine-patient-care/>
 80. KATSOUidakis, E. et al. Digital twins for health: a scoping review. In: *npj Digit. Med*. 2024, vol. 7, 77. ISSN 2398-6352.
 81. SCOTT, J. What Are Digital Twins and How Can They Be Used in Healthcare? In: *Health Tech* [online]. 2024, [cited 07.08.2025]. Available from: <https://healthtechmagazine.net/article/2024/01/what-are-digital-twins-and-how-can-they-be-used-healthcare>
 82. SHEN, M., CHEN, S., DING, X. The effectiveness of digital twins in promoting precision health across the entire population: a systematic review. In: *npj Digit. Med*. 2024, vol. 7, 145. ISSN 2398-6352.
 83. Siemens Healthineers. What is a digital patient twin? ©2024 [cited 07.08.2025]. Available from: <https://www.siemens-healthineers.com/perspectives/digital-patient-twin>
 84. HALPERN, G.A. et al. Advances and utility of digital twins in critical care and acute care medicine: a narrative review. In: *J Yeungnam Med Sci*. 2024, vol. 2025;42:9. ISSN 2799-8010.
 85. CHASE, G.J. et al. Digital Twins in Critical Care: What, When, How, Where, Why? In: *IFAC-PapersOnLine*. 2021, vol. 54 (15), pp. 310-315. ISSN 2405-8963.
 86. LI, X. et al. Digital twins as global learning health and disease models for preventive and personalized medicine. In: *Genome Med*. 2025, vol. 17, 11. ISSN 1756-994X.
 87. FORTMANN-ROE, S., BELLINGER, G. The Mathematics of Modeling: Differential Equations and System Dynamics [Systems thinking & modelling series]. 2017 [online] [cited 07.08.2025]. Available from: <https://realkm.com/2017/11/28/the-mathematics-of-modeling-differential-equations-and-system-dynamics-systems-thinking-modelling-series/>
 88. IAPASCURTA, V. Oxygen Delivery and Consumption Regulation [software] (model ID 5328). [cited 07.08.2025]. Available from: https://modelingcommons.org/browse/one_model/5328
 89. KU Libraries. The University of Kansas. Mathematical Modeling of System Dynamics – Control Systems, ©2025 [cited 07.08.2025]. Available from: <https://opentext.ku.edu/controlsystems/chapter/chapter-1/>

90. **IAPĂSCURTĂ, V.** Exploring the human circulatory system through system dynamics: a model-based approach. In: *Journal of Engineering Science*. 2024, vol. 31(4), pp. 95-104. ISSN 2587-3474.
91. US Department of Health and Human Services. HIPAA ©2025 [cited 07.08.2025]. Available from: <https://www.hhs.gov/hipaa/index.html>
92. **IAPĂSCURTĂ, V.** NetLogo integrated development environment for modeling physiological processes. In: *Conferința tehnico-științifică a studenților, masteranzilor și doctoranzilor, 27-29 martie 2024, Chișinău*. Chișinău, Republica Moldova. Tehnica-UTM, 2024, vol. 2, pp. 992-995. ISBN 978-9975-64-460-0.
93. **IAPASCURTA, V.** ICNBME Cardiac output-Simple_Model_01_01 [software] (model ID 5361) [cited 14.11.2025]. Available from: https://modelingcommons.org/browse/one_model/5361
94. DE ANDRADE L. et al. System dynamics modeling in the evaluation of delays of care in ST-segment elevation myocardial infarction patients within a tiered health system. In: *PLoS One*. 2014, vol. 31; 9(7):e103577. ISSN 1932-6203.
95. EPSTEIN, J.M. Agent-based computational models and generative social science. In: *Complexity*. 1999, vol. 4(5), pp. 41-60. ISSN 1076-2787.
96. HUNTER, P.J. et al. Modeling human physiology: The IUPS/EMBS Physiome Project. In: *Proceedings of the IEEE*. 2006, vol. 94(4), pp. 678-691. ISSN 0018-9219.
97. TRACY, M. et al. Agent-Based Modeling in Public Health: Current Applications and Future Directions. In: *Annual Review of Public Health*. 2018, vol. 39, pp. 77-94. ISSN 0163-7525.
98. NIANOGO, R.A., ARAH, O.A. Agent-based modeling of noncommunicable diseases: a systematic review. In: *American Journal of Public Health*. 2015, vol. 105(3), e20-e31. ISSN 0090-0036.
99. BADHAM, J. et al. Developing agent-based models of complex health behaviour. In: *Health Place*. 2018, vol. 54, pp. 170-177. ISSN 1353-8292.
100. ARNOLD, J.J., GAWRYS, B.L. Intrapartum Fetal Monitoring. In: *Am Fam Physician*. 2020, vol. 102(3), pp. 158-167. ISSN 0002-838X.
101. COSTA, A., COSTANTINO, M.L., FUMERO, R. Oxygen exchange mechanisms in the human placenta: mathematical modelling and simulation. In: *J Biomed Eng*. 1992, vol. 14(5):385-9. ISSN 0141-5425.
102. **IAPASCURTA, V., MUNTEANU, V., BELÎ, A.** Exploring Maternal-Placental-Fetal Interactions: A Hybrid Modeling Approach for Biomedical Engineering. In: *Sontea, V., Tiginyanu, I., Railean, S. (eds.) 7th International Conference on Nanotechnologies and Biomedical Engineering. ICNBME 2025. IFMBE Proceedings, vol. 135*. Springer, Cham. pp. 94-303. https://doi.org/10.1007/978-3-032-06497-4_30.
103. DJANATLIEV, A. et al. Hybrid Simulation with Loosely Coupled System Dynamics and Agent-Based Models for Prospective Health Technology Assessments. In: *Proceedings of the Winter Simulation Conference 2012, Berlin, Germany* Piscataway, NJ/USA: IEEE Press, 2012, pp. 770-781.
104. BRAILSFORD, S.C. et al. Hybrid simulation modelling in operational research: A state-of-the-art review. In: *European Journal of Operational Research*. 2019, vol. 278(3), pp. 721-737. ISSN 0377-2217.
105. DONG, Y. et al. Systems modeling and simulation applications for critical care medicine. In: *Ann Intensive Care*. 2012, vol. 2(1):18. ISSN 2110-5820.

106. LIU, S., TRIANTIS, K.P., ZHAO, L., WANG, Y. Capturing multi-stage fuzzy uncertainties in hybrid system dynamics and agent-based models for enhancing policy implementation in health systems research. In: *PLoS One*. 2018, vol. 13(4):e0194687. ISSN 1932-6203.
107. SERI, I., EVANS, J. Controversies in the diagnosis and management of hypotension in the newborn infant. In: *Curr Opin Pediatr*. 2001, vol. 13(2):116-23. ISSN 1040-8703.
108. CHAHAL, K., ELDABI, T., YOUNG, T. A conceptual framework for hybrid system dynamics and discrete event simulation for healthcare. In: *Journal of Enterprise Information Management*. 2013, vol. 26(1/2), pp. 50–74. ISSN 1741-0398.
109. ZENIL, H. A Review of Methods for Estimating Algorithmic Complexity: Options, Challenges, and New Directions. In: *Entropy*. 2020, vol. 22(6), 612. doi.org/10.3390/e22060612. ISSN 1099-4300.
110. WEAVER, L. et al. Digital Twins of Acute Hypoxemic Respiratory Failure Patients Suggest a Mechanistic Basis for Success and Failure of Noninvasive Ventilation. In: *Crit Care Med*. 2024, vol. 52(9):e473-e484. ISSN 1530-0293.
111. KURUPPU APPUHAMILAGE, G.D.K., HUSSAIN, M., ZAMAN, M., ALI KHAN, W. A health digital twin framework for discrete event simulation-based optimised critical care workflows. In: *NPJ Digit Med*. 2025, vol. 8(1):376. ISSN 2398-6352.
112. ROVATI, L. et al. Development and usability testing of a patient digital twin for critical care education: a mixed methods study. In: *Front Med (Lausanne)*. 2024, vol. 10:1336897. ISSN 2296-858X.
113. BRUYNSEELS, K., SANTONI DE SIO, F., VAN DEN HOVEN, J. Digital Twins in Health Care: Ethical Implications of an Emerging Engineering Paradigm. In: *Front Genet*. 2018, vol. 9:31. ISSN 1664-8021.
114. Merriam-Webster's Collegiate Dictionary (10th ed.). (1999) Merriam-Webster Inc
115. WALKER, H.K. *The origins of Physical Examination, in Clinical Methods: The History, Physical, and Laboratory Examinations*. H.K. Walker, W.D. Hall, and J.W. Hurst (eds.), 3rd edition, Boston: Butterworths, 1990. ISBN 978-0409900774.
116. BUCHARD, A. and RICHENS, J.G. Artificial Intelligence for Medical Decisions, in Springer Nature Switzerland AG 2021 N. Lidströmer, H. Ashrafian (eds.), In: *Artificial Intelligence in Medicine*, https://doi.org/10.1007/978-3-030-58080-3_28-1. ISSN 1873-2860
117. O'REGAN, G. *A Brief History of Computing*, 3rd ed. Springer, 2021. ISBN 978-3030665982
118. HAENLEIN, A., KAPLAN, A. *Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence*, California Management Review, 61(4):000812561986492, 2019. <https://doi.org/10.1177/0008125619864925>
119. VIOLATO, C. A brief history of the regulation of medical practice: Hammurabi to the national board of medical examiners. In: *J Sci Med*. 2016; 2:122–122. ISSN 2345-0592.
120. VON NEUMANN, J., MORGENSTERN, O. *Theory of games and economic behavior*. Princeton University Press; 2007. ISBN 978-0691130613.
121. LEDLEY, R.S., LUSTED, L.B. Reasoning foundations of medical diagnosis. In: *Science*. 1959; 130(3366):9–21. ISSN 0036-8075.
122. DE DOMBAL, F.T. et al. Computer-aided diagnosis of acute abdominal pain. In: *Br Med J*. 1972; 2(5804):9–13. ISSN 1756-1833.

123. SHORTLIFFE, E.H. Mycin: a knowledge-based computer program applied to infectious diseases. In: *Proceedings of the annual symposium on computer application in medical care. American Medical Informatics Association*; 1977. p. 66. ISSN 1067-5027.
124. MYERS, J.D., POPLER, H.E., MILLER, R.A. Caduceus: a computerized diagnostic consultation system in internal medicine. In: *Proceedings of the annual symposium on computer application in medical care. In: American Medical Informatics Association*; 1982. p. 44. ISSN 1067-5027. ISSN 1067-5027.
125. RAMNARAYAN, P. et al. A novel diagnostic aid (ISABEL): development and preliminary evaluation of clinical performance. In: *Medinfo*. 2004; 107:1091–5. ISSN 1569-6332.
126. IMHOFF, M., KUHLS, S. Alarm algorithms in critical care monitoring. In: *Anesth Analg*. 2006; 102(5):1525–37. ISSN 0003-2999.
127. BORTOLAN, G., DEGANI, R., WILLEMS, J.L. ECG classification with neural networks and cluster analysis. In: *Comput Cardiol*. 1991; 20:177–80. ISSN 2325-8861.
128. FIESCHI, M. et al. Medical decision support systems: old dilemmas and new paradigms? Tracks for successful integration and adoption. In: *Methods Inf Med*. 2003; 42(3):190–8. ISSN 0026-1270.
129. FURMAN, J., R. SEAMANS, R. AI and the economy. In: *Innov Policy Econ*. 2019; 19(1):161–91. ISSN 1531-3468.
130. RICHENS, J.G., BUCHARD, A. (2022). Artificial Intelligence for Medical Diagnosis. In: *Lidströmer, N., Ashrafian, H. (eds.) Artificial Intelligence in Medicine*. Springer, Cham. https://doi.org/10.1007/978-3-030-64573-1_29. ISBN 978-3-030-64572-4.
131. NEWMAN-TOKER, D.E., PRONOVOST, P.J. Diagnostic errors: the next frontier for patient safety. In: *JAMA*. 2009;301(10):1060–2. ISSN 0098-7484.
132. GRABER, M.L. The incidence of diagnostic error in medicine. In: *BMJ Qual Saf*. 2013;22(Suppl 2):ii21–7. ISSN 2044-5415.
133. SINGH, H. et al. The global burden of diagnostic errors in primary care. In: *BMJ Qual Saf*. 2017;26(6):484–94. ISSN 2044-5415.
134. NEWMAN-TOKER, D.E., MCDONALD, K.M., MELTZER, D.O. How much diagnostic safety can we afford, and how should we decide? A health economics perspective. In: *BMJ Qual Saf*. 2013;22(Suppl 2):ii11–20. ISSN 2044-5415.
135. SADEGH-ZADEH, K. Fuzzy logic. In: *Handbook of analytic philosophy of medicine*. Netherlands: Springer; 2015. p. 1055–110. ISBN 978-9-401-79578-4.
136. SAMPATH, M., LAFORTUNE, S., TENEKETZIS, D. Active diagnosis of discrete-event systems. In: *IEEE Trans Autom Control*. 1998;43(7):908–29. ISSN 1558-2523.
137. PEIRCE, C.S., *Philosophical writings of Peirce (J. Buchler, ed.)*. Vol 217. New York: Dover. 1955.
138. RAMONI, M., STEFANELLI, M., MAGNANI, L., BAROSI, G. An epistemological framework for medical knowledge-based systems. In: *IEEE Trans Syst Man Cybern*. 1992;22(6):1361–75. ISSN 2168-2232.
139. PELACCIA, T., TARDIF, J., TRIBY, E., CHARLIN, B. An analysis of clinical reasoning through a recent and comprehensive approach: the dual-process theory. In: *Med Educ Online*. 2011;16(1):5890. ISSN 1087-2981.
140. SLOMAN, S.A. The empirical case for two systems of reasoning. In: *Psychol Bull*. 1996;119(1):3. ISSN 0033-2909.

141. MAO, Q. et al. Multicentre validation of a sepsis prediction algorithm using only vital sign data in the emergency department, general ward and ICU. In: *BMJ Open*. 2018;8(1):e017833, 2018. ISSN 2044-6055.
142. HAENSSLE, H.A. et al. Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists. In: *Ann Oncol*. 2018; 29(8):1836–42. ISSN 0923-7534
143. SHWE, M.A. et al. Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. In: *Methods Inf Med*. 1991;30 (4):241–55. ISSN 0026-1270.
144. BAKATOR, M., RADOSAV, D. Deep learning and medical diagnosis: a review of literature. In: *Multimod Technol Interact*. 2018;2(3):47. ISSN 2414-4088.
145. MARCUS, G. Deep learning: a critical appraisal. arXiv preprint arXiv:1801.00631. 2018.
146. BADGELEY, M.A. et al. Deep learning predicts hip fracture using confounding patient and healthcare variables. In: *NPJ Digit Med*. 2019;2(1):1–10. ISSN 2398-6352.
147. DEGRAVE, A.J., JANIZEK, J.D., LEE, S-I. AI for radiographic COVID-19 detection selects shortcuts over signal. medRxiv. 2020.
148. GARG, A.X. et al. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. In: *JAMA*. 2005;293 (10):1223–38. ISSN 0098-7484.
149. BRUNO, M.A., WALKER, E.A., ABUJUDEH, H.H. Understanding and confronting our mistakes: the epidemiology of error in radiology and strategies for error reduction. In: *Radiographics*. 2015;35(6):1668–76. ISSN 0271-5333.
150. MCKINNEY, S.M. et al. International evaluation of an AI system for breast cancer screening. In: *Nature*. 2020;577(7788):89–94. ISSN 1476-4687.
151. ROTMENSCH, M. et al. Learning a health knowledge graph from electronic medical records. In: *Sci Rep*. 2017; 7(1):1–11. ISSN 2045-2322.
152. HECKERMAN, D.E., SHORTLIFFE, E.H. From certainty factors to belief networks. In: *Artificial Intelligence in Medicine* 4(1), 1992, pp. 35-52, [https://doi.org/10.1016/0933-3657\(92\)90036-O](https://doi.org/10.1016/0933-3657(92)90036-O) ISSN 2227-9032.
153. KINGSLAND, L.C., LINDBERG, D.A.B., SHARP, G.C. AI/RHEUM. In: *J Med Syst*. 1983;7(3):221–7. ISSN 0148-5598.
154. ADLASSNIG, K-P. et al. CADIAG: approaches to computer-assisted medical diagnosis. In: *Comput Biol Med*. 1985;15(5):315–35. ISSN 0010-4825.
155. FIESCHI, M., JOUBERT, M., FIESCHI, D., ROUX, M. SPHINX—a system for computer-aided diagnosis. In: *Methods Inf Med*. 1982;21(03):143–8. ISSN 0026-1270.
156. GODO, L.L., DE MÁNTARAS, R.L., SIERRA, C., VERDAGUER, A. Managing linguistically expressed uncertainty in milord application to medical diagnosis. In: *AI Commun*. 1(1):14–31. ISSN 1875-8452.
157. KOUR, H., MANHAS, J., SHARMA, V. Usage and implementation of neuro-fuzzy systems for classification and prediction in the diagnosis of different types of medical disorders: a decade review. In: *Artif Intell Rev*. 2020;53(7):4651–706. ISSN 0269-2821.
158. KÖHLER, S. et al. Clinical diagnostics in human genetics with semantic similarity searches in ontologies. In: *Am J Hum Genet*. 2009;85(4):457–64. ISSN 0002-9297.
159. GOUNOT, V.B. et al. Creating an ontology-driven rules base for an expert system for medical diagnosis. In: *Stud Health Technol Inform*. 2011;169:714–8. ISSN 0926-9630.

160. BODENREIDER, O. The Unified Medical Language System (UMLS): integrating biomedical terminology. In: *Nucleic Acids Res.* 2004; 32(suppl 1): D267–70. ISSN 0305-1048.
161. DAVIS, R. Expert systems: where are we? And where do we go from here? In: *AI Mag.* 1982; 3(2): 3–3. ISSN 0738-4602.
162. BYLANDER, T. Some causal models are deeper than others. In: *Artif Intell Med.* 1990; 2(3):123–8. ISSN 2227-9032.
163. POOLE, D. Normality and faults in logic-based diagnosis. In: *IJCAI*, vol. 89. Citeseer; 1989. p. 1304–10. ISSN 1045-0823.
164. COX, P.T., PIETRZYKOWSKI, T. General diagnosis by abductive inference. In: *SLP*, vol. 183. 1987. p. 189. ISSN 1932-4731.
165. WEISS, S.M., KULIKOWSKI, C.A., AMAREL, S., SAFIR, A. A model-based method for computer-aided medical decision-making. In: *Artif Intell.* 1978; 11(1–2):145–72. ISSN 0004-3702.
166. REGGIA, J.A., NAU, D.S., WANG, P.Y. A formal model of diagnostic inference. I. Problem formulation and decomposition. In: *Inf Sci.* 1985;37(13):227–56. ISSN 0020-0255.
167. MANI, N., SLEVIN, N., HUDSON, A. What three wise men have to say about diagnosis. In: *BMJ.* 2011;343:d7769. ISSN 1756-1833.
168. PEARL, J. *Causality (2nd ed.)*. Cambridge: Cambridge University Press; 2009. ISBN 978-0521895606.
169. RICHENS, J.G., LEE, C.M., JOHRI, S. Improving the accuracy of medical diagnosis with causal machine learning. In: *Nat Commun.* 2020;11(1):1–9. ISSN 2041-1723.
170. YOON, J.H., PINSKY, M.R., and CLERMONT, G. Artificial Intelligence in Critical Care Medicine. In: *Critical Care* (2022) 26:75 <https://doi.org/10.1186/s13054-022-03915-3>. ISSN 1364-8535.
171. SEAH, J.C.Y. et al. Chest radiographs in congestive heart failure: visualizing neural network learning. In: *Radiology.* 2019;290:514–22. ISSN 0033-8419.
172. VINCENT, J.L. The continuum of critical care. In: *Crit Care.* 2019;23(Suppl 1):122. ISSN 1364-8535.
173. CHEN, L. et al. Dynamic and personalized risk forecast in step-down units. Implications for monitoring paradigms. In: *Ann Am Thorac Soc.* 2017;14:384–91. ISSN 2329-6933.
174. YOON, J.H. et al. Predicting tachycardia as a surrogate for instability in the intensive care unit. In: *J Clin Monit Comput.* 2019;33:973–85. ISSN 1573-2614.
175. WIJNBERGE, M. et al. Effect of a machine learning-derived early warning system for intraoperative hypotension vs standard care on depth and duration of intraoperative hypotension during elective noncardiac surgery: the HYPE randomized clinical trial. In: *JAMA.* 2020;323:1052–60. ISSN 0098-7484.
176. SEYMOUR, C.W. et al. Derivation, validation, and potential treatment implications of novel clinical phenotypes for sepsis. In: *JAMA.* 2019;321:2003–17. ISSN 0098-7484.
177. ROWAN, K.M. et al. Early, goal-directed therapy for septic shock—a patient-level meta-analysis. In: *N Engl J Med.* 2017;376:2223–34. ISSN 0028-4793.

178. KOMOROWSKI, M. et al. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. In: *Nat Med*. 2018;24:1716–20. ISSN 1078-8956.
179. PEINE, A. et al. Development and validation of a reinforcement learning algorithm to dynamically optimize mechanical ventilation in critical care. In: *NPJ Digit Med*. 2021;4:32. ISSN 2398-6352.
180. Epilepsy: a public health imperative. Geneva: World Health Organization; 2019. ISBN 978-92-4-151593-1.
181. USMAN, S. M. et al. Detection of preictal state in epileptic seizures using ensemble classifier. In: *Epilepsy Research*, Vol. 178, 2021. ISSN 0920-1211.
182. WEI, X. et al. Early prediction of epileptic seizures using a long-term recurrent convolutional network. In: *Journal of Neuroscience Methods*, vol. 327, 2019. ISSN 0165-0270.
183. PUTHANKATTIL SUBHA, D., JOSEPH, P. K., RAJENDRA ACHARYA U., and CHOO MIN LIM, EEG Signal Analysis: A Survey. In: *Journal of Medical Systems*, vol. 34(), 2010, pp. 195–212. ISSN 0148-5598.
184. ZENIL, H. Towards Demystifying Shannon Entropy, Lossless Compression and Approaches to Statistical Machine Learning. In: *Proceedings 2020*, 47, 24. <https://doi.org/10.3390/proceedings2020047024>. ISSN 2504-3900.
185. SINGER M., DEUTSCHMAN C., SEYMOUR C. et al. The third International consensus definitions for sepsis and septic shock (sepsis-3). In: *JAMA*. 2016, 315:801–10. ISSN 0098-7484.
186. GEROULANOS, S., DOUKA, T. Historical perspective of the word “sepsis”. In: *Intensive care medicine*. 2006, 32:12, pp. 2077–2087. ISSN 0342-4642.
187. 175. BONE, R. et al. Definitions for sepsis and organ failure and guidelines for the use of innovative therapies in sepsis. In: *Chest*. 1992, 101:6, pp. 1644–1655. ISSN 0012-3692.
188. LEVY, M. et al. 2001 SCCM/ESICM/ACCP/ATS/SIS International Sepsis Definitions Conference. In: *Intensive Care Medicine*. 2003, 29:4, pp. 530–538. ISSN 0342-4642.
189. VINCENT, J. et al. The SOFA (Sepsis-related Organ Failure Assessment) score to describe organ dysfunction/failure. In: *Intensive Care Medicine*. 1996, 22:7, pp. 707–710. ISSN 0342-4642.
190. SKINNER, Q. *Machiavelli: The Prince*. 1988. ISBN 978-0521349932.
191. FERRER, R. et al. Empiric antibiotic treatment reduces mortality in severe sepsis and septic shock from the first hour: results from a guideline-based performance improvement program. In: *Critical Care Medicine*. 2014, 42:8, pp. 1749–1755. ISSN 0090-3493.
192. HOTCHKISS, R. et al. Sepsis and septic shock. In: *Nature Reviews Disease Primers* 2, 2016. ISSN 2056-676X.
193. KAUKONEN, R. et al. Mortality related to severe sepsis and septic shock among critically ill patients in Australia and New Zealand, 2000-2012. In: *JAMA* 2014 311(13):1308-16. doi: 10.1001/jama.2014.2637. ISSN 0098-7484.
194. RHODES, A. et al. Surviving sepsis campaign: international guidelines for management of sepsis and septic shock: 2016. In: *Intensive Care Medicine*. 2017, 43:3, pp. 304–377. ISSN 0342-4642.

195. MARIK, P., FARKAS, D. The changing paradigm of sepsis: early diagnosis, early antibiotics, early pressors, and early adjuvant treatment. In: *Critical care medicine*. 2018, 46:10, pp. 1690–1692. ISSN 0090-3493.
196. BURDICK, H. et al. Effect of a sepsis prediction algorithm on patient mortality, length of stay and readmission: a prospective multicentre clinical outcomes evaluation of real-world patient data from US hospitals. In: *BMJ Health Care Inform*. 2020, 27:e100109. doi:10.1136/bmjhci-2019-100109. ISSN 2632-1009.
197. DELAHANTY R., ALVAREZ J., FLYNN L. et al. Development and evaluation of a machine learning model for the early identification of patients at risk for sepsis. In: *Ann Emerg Med*. 2019, 73:334–44. ISSN 0196-0644.
198. CUNNINGHAM, F. et al. Ensembl 2019. In: *Ensembl Nucleic Acids Res* 2019;47(D1):2019. ISSN 0305-1048.
199. ZOU, J. et al. A primer on deep learning in genomics. In: *Nat Genet* 2019; 51(1):12. ISSN 1061-4036.
200. CHOWDHURY, G.G. Natural language processing. In: *Annu Rev Inf Sci Technol* 2005; 37(1):51–89. <https://doi.org/10.1002/aris.1440370103>. ISSN 0066-4200.
201. IUCHI, H. et al. Representation learning applications in biological sequence analysis. In: *Computational and Structural Biotechnology Journal*, 19 (2021), 3198-3208. ISSN 2001-0370.
202. YU, L. et al. Grammar of protein domain architectures. In: *Proc Natl Acad Sci USA* 2019; 116(9):3636–45. ISSN 0027-8424.
203. BENGIO, Y., COURVILLE, A., VINCENT, P. Representation learning: a review and new perspectives. In: *IEEE Trans Pattern Anal Mach Intell* 2013;35(8):1798–828. <https://doi.org/10.1109/TPAMI.2013.50>. arXiv:1206.5538.
204. RAO, R. et al. Transformer protein language models are unsupervised structure learners. In: *International Conference on Learning Representations (2021)*. <https://openreview.net/pdf?id=fylclEqgvvd>
205. LITTMANN, M. et al. Embeddings from deep learning transfer GO annotations beyond homology. In: *Sci Rep* 2021;11(1):1160. <https://doi.org/10.1038/s41598-020-80786-0>. ISSN 2045-2322.
206. KIM, Y. et al. Combining Representation Learning with Tensor Factorization for Risk Factor Analysis—an application to Epilepsy and Alzheimer’s disease. arXiv:1905.05830.
207. ZHANG, X. et al. Adversarial Representation Learning for Robust Patient-Independent Epileptic Seizure Detection, 2015. arXiv:1909.10868.
208. CHOI, E. et al. Multi-layer Representation Learning for Medical Concepts. In: *KDD '16*, August 13-17, 2016, San Francisco, CA, USA. ISBN 978-1-4503-4232-2.
209. BANVILLE, H. et al. Self-supervised representation learning from electroencephalography signals, arXiv:1911.05419v1, 2019
210. REZAIL, T.Y., MOUSAVI, Z., DELPAK, A., FARZAMINA, A. Automatic Identification of Epileptic Seizures From EEG Signals Using Sparse Representation-Based Classification, In: *IEEE Access*, 2020. ISSN 2169-3536.
211. WANG, J. et al. A Weighted Overlook Graph Representation of EEG Data for Absence Epilepsy Detection, In: *2020 IEEE International Conference on Data Mining (ICDM)*, 17–20 November, 2020. ISBN 978-1-7281-8316-9.

212. MOHSENVAND, M.N., IZADI, M.R. Contrastive Representation Learning for Electroencephalogram Classification. In: *Proceedings of Machine Learning Research* 136:238–253, 2020. ISSN 2640-3498.
213. WEI, R., MAHMOOD, A. Recent Advances in VAEs With Representation Learning for Biomedical Informatics: A Survey, In: *IEEE Access*, 2020. ISSN 2169-3536.
214. RAMOS, G., GJINI, E., COELHO, L., and SILVEIRA, M. Unsupervised learning approach for predicting sepsis onset in ICU patients, In: *43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, Oct 31 - Nov 4, 2021. ISBN 978-989-758-631-6.
215. ROCHETEAU, E. et al. Predicting Patient Outcomes with Graph Representation Learning, arXiv:2101.03940v1 [cs.LG], 2021.
216. TRAN, L., NGUYEN, M., SHAHABI, C. Representation Learning for Early Sepsis Prediction, In: *Computing in Cardiology*, vol 46, 2019. ISSN 2325-8861.
217. KILLIAN, T. W. et al. An Empirical Study of Representation Learning for Reinforcement Learning in Healthcare, arXiv:2011.11235v1 [cs.LG], 2020.
218. STEINBERG, E. et al. Language models are an effective representation learning technique for electronic health record data. In: *Journal of Biomedical Informatics*, Vol. 113, 103637, 2021. ISSN 1532-0464.
219. CHEN, Y., QI, B. Representation learning in intraoperative vital signs for heart failure risk prediction. In: *BMC Medical Informatics and Decision Making*, 19:260, 2019. ISSN 1472-6947.
220. ZHU, X. et al. Embedding, aligning and reconstructing clinical notes to explore sepsis. In: *BMC Res Notes*, 14:136, 2021. ISSN 1756-0500.
221. ZENIL, H. Causal Model discovery with Algorithmic Information Dynamic, Jul 3 2023 [cited 10.12.2023]. Available: <https://www.youtube.com/watch?v=-70R37nR3AY>
222. WANG, M., YIN, X., ZHU, Y. and HU, J. Representation Learning and Pattern Recognition in Cognitive Biometrics: A Survey. In: *Sensors* 2022, 22, 5111. <https://doi.org/10.3390/s22145111>. ISSN 1424-8220.
223. WANG, M., HU, J., ABBASS, H.A. Multi-scale weighted inherent fuzzy entropy for EEG biomarkers. In: *Proceedings of the 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8. ISBN 978-1-5090-6020-7.
224. RAHMAN, M.W., GAVRILOVA, M.L. Emerging EEG and kinect face fusion for biometric identification. In: *Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI)*, Honolulu, HI, USA, 27 November–1 December 2017; pp. 1–8. ISBN 978-1-5386-2727-3.
225. THOMAS, K.P., VINOD, A.P. Biometric identification of persons using sample entropy features of EEG during rest state. In: *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Budapest, Hungary, 9–12 October 2016; pp. 3487–3492. ISBN 978-1-5090-1897-0.
226. MU, Z., HU, J., MIN, J. EEG-based person authentication using a fuzzy entropy-related approach with two electrodes. In: *Entropy* 2016, 18, 432. ISSN 1099-4300.
227. CHEN, W., WANG, Z., XIE, H., YU, W. Characterization of surface EMG signal based on fuzzy entropy. In: *IEEE Trans. Neural Syst. Rehabil. Eng.* 2007, 15, 266–272. ISSN 1534-4320.

228. CAO, Z., LIN, C.T. Inherent fuzzy entropy for the improvement of EEG complexity evaluation. In: *IEEE Trans. Fuzzy Syst.* 2017, 26, 1032–1035. ISSN 1063-6706.
229. ZHENG, W.L. et al. Emotionmeter: A multimodal framework for recognizing human emotions. In: *IEEE Trans. Cybern.* 2018, 49, 1110–1122. ISSN 2168-2267.
230. ZHENG, W.L., LU, B.L. Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks. In: *IEEE Trans. Auton. Ment. Dev.* 2015, 7, 162–175. ISSN 1943-0604.
231. LIN, S. et al. Individual identification based on chaotic electrocardiogram signals during muscular exercise. In: *IET Biom.* 2014, 3, 257–266. ISSN 2047-4938.
232. YASAKA, K., ABE, O. Deep learning and artificial intelligence in radiology: Current applications and future directions. In: *PLoS Med.* 2018, 15(11): e1002707. <https://doi.org/10.1371/journal.pmed.1002707>. ISSN 1549-1676.
233. NIAZI KHAN, K.M. et al. Digital pathology and artificial intelligence. In: *The Lancet Digital Oncology*, 2019, 20(5), e253-e261. ISSN 1470-2045.
234. JOHNSON, K.W. et al. Artificial Intelligence in Cardiology. In: *Journal of the American College of Cardiology*. 2018, 71(23), doi: 10.1016/j.jacc.2018.03.521. ISSN 0735-1097.
235. PADHY, S.K. et al. Artificial intelligence in diabetic retinopathy: A natural step to the future. In: *Indian J Ophthalmol.* 2019, 67(7):1004-1009. doi: 10.4103/ijo.IJO_1989_18. ISSN 0301-4738.
236. FARLEY, M. et al. Improving the efficiency of the operating room environment with an optimization and machine learning model. In: *Health Care Manag Sci.* 2019, 22(4):756-767. doi: 10.1007/s10729-018-9457-3. ISSN 1572-9389.
237. HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd ed.*, Springer, 2017. ISBN 978-0387848570.
238. JAMES, G., WITTEN, D., HASTIE, T., TIBSHIRANI, R. *An Introduction to Statistical Learning: with Applications in R (Springer Texts in Statistics), 2nd ed.*, Springer, 2021. ISBN 978-1461471370.
239. HALL, P. and GILL, N. *An Introduction to Machine Learning Interpretability, 2nd ed.* 2019 O'Reilly Media, Inc. ISBN 9781098115470.
240. ALY, M. Survey on Multiclass Classification Methods. In: *Neural Netw.* 2005. ISSN 0893-6080.
241. Course 9.520 on Statistical Learning Theory and Applications, Instr: T. Poggio, L. Rosasco, C. Frogner, G.D. Canas, Stanford University, ppt, 2012
242. VAPNIK, V. Principles of Risk Minimization for Learning Theory. In: *Advances in neural information processing*, 1991. ISSN 1049-5258.
243. MALAKAR, S. Computer-Based Diagnosis of Some Chronic Diseases: A Medical Journey of the Last Two Decades. In: *Archives of Computational Methods in Engineering*, 2022. ISSN 1134-3060.
244. BHVASAR, K.A. et al. Medical Diagnosis Using Machine Learning: A Statistical Review. In: *Computers, Materials & Continua*, 2021. ISSN 1546-2218.
245. AHSAN, M.M., LUNA, S.A., SIDDIQUE, Z. Machine-Learning-Based Disease Diagnosis: A Comprehensive Review. In: *Healthcare*, 2022. ISSN 2227-9032.

246. SHOEIBI, A. et al. Epileptic Seizures Detection Using Deep Learning Techniques: A Review. In: *International Journal of Environmental Research and Public Health*, 18, 5780, 2021. ISSN 1660-4601.
247. CAO, J., ZHU, J., HU, W., KUMMERT, A. Epileptic Signal Classification with Deep EEG Features by Stacked CNNs. In: *IEEE Trans. Cogn. Dev. Syst.* 2020, 12, 709–722. ISSN 2379-8939.
248. ZHAO, X. et al. Classification of Epileptic EEG Signals by CNN and Data Augmentation. In: *Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Barcelona, Spain, 4–8 May 2020; pp. 926–930. ISBN 978-1-5090-6631-5.
249. ABIYEV, R. et al. Identification of epileptic eeg signals using convolutional neural networks. In: *Appl. Sci.* 2020, 10, 4089. ISSN 2076-3417.
250. GENG, M. et al. Epileptic Seizure Detection Based on Stockwell Transform and Bidirectional Long Short-Term Memory. In: *IEEE Trans. Neural Syst. Rehabil. Eng.* 2020, 28, 573–580. ISSN 1534-4320.
251. LIANG, W., PEI, H., CAI, Q., WANG, Y. Scalp EEG epileptogenic zone recognition and localization based on long-term recurrent convolutional network. In: *Neurocomputing* 2020, 396, 569–576. ISSN 0925-2312.
252. LEKSHMY, H. et al. Comparative analysis of multiple machine learning algorithms for epileptic seizure prediction. 2022. In: *J. Phys. Conf. Ser.* 2161 012055 DOI 10.1088/1742-6596/2161/1/012055. ISSN 1742-6596.
253. CHANDU, Y. S. and FATHIMABI, S. Epilepsy prediction using deep learning. In: *International Journal of Engineering Research & Technology (IJERT)*, vol. 9, no. 5, pp. 211–219, 2021. ISSN 2278-0181.
254. EPMOGHADDAM, D. et al. Epileptic seizure prediction using spectral width of the covariance matrix. In: *J. Neural Eng.* 2022, 19, 026029. ISSN 1741-2560.
255. NAZARI, J. et al. Epilepsy seizure prediction with few-shot learning method. In: *Brain Inf.* 9, 21, 2022. <https://doi.org/10.1186/s40708-022-00170-8> ISSN 2198-4018
256. YAN, J. et al. Seizure Prediction Based on Transformer Using Scalp Electroencephalogram. In: *Appl. Sci.* 2022, 12, 4158. ISSN 2076-3417.
257. MIR, W.A., ANJUM, M., SHAHAB, S. Deep-EEG: An Optimized and Robust Framework and Method for EEG-Based Diagnosis of Epileptic Seizure. In: *Diagnostics* 2023, 13, 773. <https://doi.org/10.3390/diagnostics13040773>. ISSN 2075-4418.
258. XU, X. et al. Patient-specific method for predicting epileptic seizures based on DRSN-GRU. In: *Biomedical Signal Processing and Control* 81, 2023, 104449. ISSN 1746-8108.
259. IBRAHIM, A.K. et al. Epileptic seizure prediction based on multiresolution convolutional neural networks. In: *Front. Sig. Proc.* 2023. 3:1175305. doi: 10.3389/frsip.2023.1175305. ISSN 2345-5463.
260. QIAO, W. et al. Epilepsy Prediction and Detection Using Attention-CssCDBN with Dual-Task Learning. In: *Sensors (Basel)*. 2024 Dec 25;25(1):51. doi: 10.3390/s25010051. ISSN 1424-8220.
261. LUCASIUS, C. et al. Biomimetic Deep Learning Networks With Applications to Epileptic Spasms and Seizure Prediction. *IEEE Trans Biomed Eng.* 2024 Mar;71(3):1056-1067. doi: 10.1109/TBME.2023.3325762. ISSN 0018-9294.

262. ZHANG F, ZHANG X. Prediction of Epilepsy Seizure Based on Cepstrum Analysis and Deep Learning. In: *Interdiscip Sci*. 2025 Jul 16. doi: 10.1007/s12539-025-00741-3. ISSN 2594-3405.
263. XIANG, J. et al. Synchronization-based graph spatio-temporal attention network for seizure prediction. In: *Sci Rep*. 2025 Feb 3;15(1):4080. doi: 10.1038/s41598-025-88492-5. ISSN 2045-2322.
264. WICKRAMARATNE, D., MAHMUD, S. Bi-Directional Gated Recurrent Unit Based Ensemble Model for the Early Detection of Sepsis, In: *Annu Int Conf IEEE Eng Med Biol Soc*, 2020, pp.70-73. doi: 10.1109/EMBC44109.2020.9175223. ISSN 2375-7477.
265. KOK, C. et al. Automated prediction of sepsis using temporal convolutional network. In: *Comput Biol Med*, 2020, 127:103957. doi: 10.1016/j.combiomed.2020.103957. ISSN 0010-4825.
266. HELGUERA-REPETTO, A. et al., Neonatal Sepsis Diagnosis Decision-Making Based on Artificial Neural Networks, In: *Frontiers in Pediatrics*, 8:525, 2020, <https://doi.org/10.3389/fped.2020.00525>. ISSN 2296-2360.
267. MISRA, D. et al. Early Detection of Septic Shock Onset Using Interpretable Machine Learners. In: *J Clin Med*, 2021, 10, 301. <https://doi.org/10.3390/jcm10020301>. ISSN 2077-0383.
268. SADASIVUN, S. et al. Fusion of fully integrated analog machine learning classifier with electronic medical records for real-time prediction of sepsis onset. In: *Nature Scientific Reports*, 2022, 12:5711, <https://doi.org/10.1038/s41598-022-09712>. ISSN 2045-2322.
269. KE L. et al. Identification of potential diagnostic and prognostic biomarkers for sepsis based on machine learning. In: *Computational and Structural Biotechnology Journal* 21 2023. 2316–2331. ISSN 2001-0370.
270. LAXMI L. et al. Machine Learning framework to predict sepsis disease using stacking ensemble algorithm. In: *Research Square*, 2023, <https://doi.org/10.21203/rs.3.rs-2946930/v1>. ISSN 2693-5015.
271. DALAL, S. et al. Time-Series Deep Learning and Conformal Prediction for Improved Sepsis Diagnosis in Non-ICU Hospitalized Patients. medRxiv preprint doi: <https://doi.org/10.1101/2024.11.21.24317716>.
272. UPADHYAYA, D.P. et al. Machine Learning Interpretability Methods to Characterize the Importance of Hematologic Biomarkers in Prognosticating Patients with Suspected Infection. In: *Computers in Biology and Medicine* 2024;183:109251.ISSN 0010-4825.
273. SIRI, D. et al. Bio inspired feature selection and graph learning for sepsis risk stratification. In: *Sci Rep*. 2025 May 22;15(1):17875. doi: 10.1038/s41598-025-02889-w. ISSN 2045-2322.
274. EVANS, L. et al. Surviving Sepsis Campaign: International Guidelines for Management of Sepsis and Septic Shock 2021, In: *Intensive Care Medicine*, 2021, doi:<https://doi.org/10.1007/s00134-021-06506-y>. ISSN 0342-4642.
275. ZYMA, I. et al. Electroencephalograms during Mental Arithmetic Task Performance. In: *Data*, 2019, 4(1), 14. <https://doi.org/10.3390/data4010014>. ISSN 2306-5729. 167.
276. ISMAIL, A.R., ABDIN, N.Z., MAEN, M.K. Systematic Review on Missing Data Imputation Techniques with Machine Learning Algorithms for Healthcare. In: *JRC* 2022, 3(2), pp. 143-152. ISSN 2715-5072.

277. OLUWASEYE, L.J., DOORSAMY, W., SENA, B.P. A Review of Missing Data Handling Techniques for Machine Learning. In: *IJITIS 2022*, 5(3), pp. 971-1005. ISSN 2613-7305.
278. IAPĂSCURTĂ, V., FIODOROV, I. New approaches to missing biomedical data recovery for machine learning. In: *Journal of Engineering Science*. 2023, 30(1), pp. 106-117. [https://doi.org/10.52326/jes.utm.2023.30\(1\).09](https://doi.org/10.52326/jes.utm.2023.30(1).09). ISSN 2312-2498.
279. STERNE, J.A. et al. Multiple imputation for missing data in epidemiological and clinical research: potential and pitfalls. In: *BMJ* 2009, 338:b2393. ISSN 1756-1833.
280. JAIN, R.; XU, W. Dynamic model updating (DMU) approach for statistical learning model building with missing data. In: *BMC Bioinform* 2021, 22, 221. ISSN 1471-2105.
281. EMA Guideline on Missing Data in Confirmatory Clinical Trials, 2010 [cited 14.02.2023] Available: https://www.ema.europa.eu/en/documents/scientific-guideline/guideline-missing-data-confirmatory-clinical-trials_en.pdf.
282. STREINER, D.L. Last observation Carried Forward. In: *Encyclopedia of Research Design, 1st ed.; Salkind, N.J. Ed.; SAGE Publications, Inc., 2010, 1776 p. ISBN 1412961270*.
283. ZEILEIS, A., GROTHENDIECK, G. zoo: S3 Infrastructure for Regular and Irregular Time Series. In: *J. Stat. Soft.* 2005, 14(6), pp. 1–27. ISSN 1548-7660.
284. KALMAN, R.E. A New Approach to Linear Filtering and Prediction Problems, Transactions of the ASME. In: *J. Basic Eng.* 1960, 82 (Series D), pp. 35-45. ISSN 0021-9223.
285. GOVAERS, F. *Introduction and Implementation of the Kalman Filter*. IntechOpen 2019, 128 p., doi: 10.5772/intechopen.75731. ISBN 978-1-8388-0537-1.
286. MORITZ, S., BARTZ-BEIELSTEIN, T. ImputeTS: Time Series Missing Value Imputation in R. In: *The R Journal* 2017, 9(1), pp. 207-218. ISSN 2073-4859.
287. NYKODYM, T., KRALJEVIC, T., WANG, A., WONG, W. *Generalized Linear Modeling with H2O, 6th ed.; H2O.ai, Inc.: Mountain View, CA, 2022; pp. 10-17*. [cited 07.03.2023]. Available: <https://docs.h2o.ai/h2o/latest-stable/h2o-docs/booklets/GLMBooklet.pdf>.
288. H2O.ai. (3.32.1.1, 2021). h2o: R Interface for H2O. [cited 07.03.2023]. Available: <http://h2o-release.s3.amazonaws.com/h2o/rel-zipf/1/index.html>.
289. MALOHLAVA, M., CANDEL, A. *Gradient Boosting Machine with H2O, 7th ed.; H2O.ai, Inc.: Mountain View, CA, 2022; pp. 8-14*. [cited 07.03.2023]. Available: <https://docs.h2o.ai/h2o/latest-stable/h2o-docs/booklets/GBMBooklet.pdf>.
290. ZENIL, H., SOLER-TOSCANO, F., GAUVRIT, N., KIANI, N. A., SINGMANN, H., SCOTT, G., HERNÁNDEZ, S., RUEDA-TOICEN, A., ROBERTSON, Z. and ROIG, G. G. The Online Algorithmic Complexity Calculator (OACC) v3.0, Algorithmic Dynamics Lab, Science for Life Laboratory (SciLifeLab), Unit of Computational Medicine, Center for Molecular Medicine at the Karolinska Institute in Stockholm, Sweden [software]. Oct 2018. [cited 10.12.2023]. Available: <https://complexity-calculator.com/>
291. IAPĂSCURTĂ, V. Block Decomposition Method and Traditional Machine Learning for Epileptic Seizure Prediction. In: *26th International Workshop on Cellular Automata and Discrete Complex Systems AUTOMATA 2020*, Stockholm, Sweden (2020). [online]. Available: <https://www.automata2020.com/videos-material.html>.
292. IAPASCURTA, V. Combining Algorithmic Information Dynamics Concepts and Machine Learning for Electroencephalography Analysis: What Can We Get?. In: *Complex*

- Systems* 31(4), pp. 389–413 (2022). <https://doi.org/10.25088/ComplexSystems.31.4.389>. ISSN 0891-2513.
293. **IAPASCURTA, V.** Prezicerea timpurie a sepsisului cu ajutorul unei aplicații proprii elaborate în baza învățării automate (inteligență artificială): tz. de doct. în științe medicale. Chișinău, 2023. 170 p. Available from: <https://repository.usmf.md/handle/20.500.12710/25122>.
 294. GHODSI, A. Dimensionality Reduction. A Short Tutorial. Department of Statistics and Actuarial Science, University of Waterloo, Ontario, Canada, 2006, [online]. [cited 7.10.2026]. Available at: <https://people.computing.clemson.edu/~ekp/courses/cpsc9500/assets/pca2.pdf>
 295. MORIN, F., BENGIO, Y. Hierarchical probabilistic neural network language model. In: *Proceedings of the 10th International Workshop on AI and Statistics. Proceedings of Machine Learning Research*, vol. R5, 2005, pp. 246-252. ISSN 2640-3498.
 296. MNIH, A., HINTON, G. A Scalable Hierarchical Distributed Language Model. In: *Advances in Neural Information Processing Systems 21 (NIPS 2008)*, Curran Associates Inc. 21, 2009, pp. 1081-1088. ISSN 1049-5258.
 297. HINTON, G. Learning distributed representations of concepts. In: *Proc. 8th Annu. Conf. Cogn. Sci. Soc.*, vol. 1, Aug. 1986, p. 12. ISSN 1069-7977.
 298. BENGIO, Y., DUCHARME, R., VINCENT, P., and JAUVIN, C. A neural probabilistic language model. In: *J. Mach. Learn. Res.*, vol. 3, no. 6, 2003, pp. 932–938. ISSN 1532-4435.
 299. MIKOLOV, T., SUTSKEVER, I., CHEN, K., CORRADO, G. S., and DEAN, J. Distributed representations of words and phrases and their compositionality. In: *Proc. Adv. Neural Inf. Process. Syst.*, Nevada, NV, USA, 2013, pp. 3111–3119. ISBN 978-1-6326-6024-4.
 300. MIKOLOV, T. et al. Efficient Estimation of Word Representations in Vector Space. arXiv: 1301.3781v3, 7 Sep, 2013.
 301. PENNINGTON, J., SOCHER, R., and MANNING, C. Glove: Global Vectors for Word Representation. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, vol. 31, no. 6, pp. 1532–1543, doi: 10.3115/v1/D14-1162. ISBN 978-1-6343-9478-9.
 302. BOJANOWSKI, P., GRAVE, E., JOULIN, A., and MIKOLOV, T. Enriching Word Vectors with Subword Information. In: *Transactions of the Association for Computational Linguistics*, Dec. 2017, vol. 5, pp. 135–146, doi: 10.1162/tac1_a_00051. ISSN 2307-387X.
 303. JOULIN, A., GRAVE, E., BOJANOWSKI, P., and MIKOLOV, T. Bag of tricks for efficient text classification. In: *15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference*, 2017, vol. 2, pp. 427–431, doi: 10.18653/v1/e17-2068. ISBN 978-1-945626-35-7.
 304. DEVLIN, J., CHANG, M-W., LEE, K., and TOUTANOVA, K. BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805v2, May 2019
 305. VASWANI, A. et al. Attention is all you need. In: *I. Guyon et al. (eds), Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2017. arXiv:1706.03762.

306. SHU-HENG, W. A., IBRAHIM, T. U., and ABIDEREXITI, K. Sentiment classification of Uyghur text based on BLSTM. In: *Comput. Eng. Des.*, vol. 38, no. 10, pp. 2879–2886, 2017. ISSN 2288-5048.
307. XU, G. et al. Sentiment Analysis of Comment Text Based on BiLSTM, In: *IEEE Access*, vol 7, 2019, pp. 51522–51532. ISSN 2169-3536.
308. **IAPASCURTA V.** Some aspects of deep representation learning on transformed EEG signal data. In: *Technical Scientific Conference of Undergraduate, Master and PhD Students, Technical University of Moldova*. Chisinau, April 5-7, 2023, Vol. I
309. REHUREK, R., SOJKA, P. Gensim—python framework for vector space modelling. NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic. 2011; 3(2).
310. MCINNES, L., HEALY, J., MELVILLE, J. UMAP: Uniform Manifold Approximation for Dimension Reduction. arXiv:1802.03426v3, Sep 2020. https://umap-learn.readthedocs.io/en/latest/basic_usage.html#
311. Scikit-learn: Machine Learning in Python, Pedregosa et al., In: *JMLR 12*, pp. 2825–2830, 2011. ISSN 1533-7928.
312. PEARSON, K. On lines and planes of closest fit to systems of points in space. In: *Philos Mag.* 1901; 2:559e572. ISSN 1478-6443.
313. FISHER, R. The use of multiple measurements in taxonomic problems. In: *Ann Eugen.* 1936; 7:179e188. ISSN 0003-4800.
314. SCHÖLKOPF, B., SMOLA, A., MULLER, K-R. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. In: *Neural Computation*. 10 (5): 1299–1319. ISSN 0899-7667.
315. TIPPING, M., BISHOP, C. Probabilistic Principal Component Analysis. In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, Vol. 61, No. 3, 1999, pp. 611–622. ISSN 1369-7412.
316. ERICSSON, L., GOUK, H., CHANGE, L.C., HOSPEDALES, T.M. Self-Supervised Representation Learning: Introduction, advances and challenges. In: *IEEE Signal Processing Magazine*. 39 (3), 2020, pp. 42–62. ISSN 1558-0792.
317. **IAPASCURTA, V.** A less traditional approach to biomedical signal processing for sepsis prediction. In: *5th International Conference on Nanotechnologies and Biomedical Engineering, Chisinau, Moldova, November 3-5, 2021; Tiginyanu, Sontea, Railean, Eds.; Springer IFMBE Proceedings Series*, 2022; pp. 215-222. ISSN 1680-0737.
318. **IAPASCURTA, V.** Dealing with Missing Continuous Biomedical Data: a Data Recovery Method for Machine Learning Purposes. In: *The 12th International Conference on Electronics, Communications and Computing*, Chisinau, Moldova, October 20-21, 2022.
319. BOUWMEESTER, W. et al. Reporting and methods in clinical prediction research: a systematic review. In: *PLoS Med* 2012;9:e1001221. ISSN 1549-1277.
320. COLLINS, G. et al. Protocol for development of a reporting guideline (TRIPOD-AI) and risk of bias tool (PROBAST-AI) for diagnostic and prognostic prediction model studies based on artificial intelligence. In: *BMJ Open* 2021; 11:e048008, pp. 1-7. ISSN 2398-8703.
321. FAES, L. et al. A clinician's guide to artificial intelligence: how to critically appraise machine learning studies. In: *Transl Vis Sci Technol* 2020;9:7. ISSN 2164-2591.
322. BOSSUYT, P.M. et al. STARD 2015: an updated list of essential items for reporting diagnostic accuracy studies. In: *BMJ* 2015;351:h5527, pp. 1-9. ISSN 1756-1833.

323. SOUNDERAJAH, V. et al. Developing a reporting guideline for artificial intelligence-centred diagnostic test accuracy studies: the STARD-AI protocol On behalf of the STARD-AI Steering Committee. In: *BMJ Open*, (11), 2021, 47709 pp. 1-7. ISSN 2398-8703.
324. MedCalc Software Ltd. Diagnostic test evaluation calculator. [software]. Version 22.014 [cited 14.11.23]. Available: https://www.medcalc.org/calc/diagnostic_test.php.
325. SAITO, T., REHMSMEIER, M. The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. In: *PLoS ONE* 10(3): e0118432, 2015. doi:10.1371/journal.pone.0118432. ISSN 1932-6203.
326. BertViz. Visualize Attention in NLP Models. [cited 14.11.23]. Available: <https://github.com/jessevig/bertviz#head-and-model-views>.
327. IAPĂSCURTĂ V., BELÎ A. Preclinical stage of building a machine learning system for sepsis prediction: a comparative study of four algorithms. In: *5th International Conference on Nanotechnologies and Biomedical Engineering, November 3-5, 2021*, Springer IFMBE Proceedings Series, 2022, pp. 448-455. ISSN 1680-0737.
328. GUYON, I. and ELISSEEFF, A. An introduction to variable and feature selection. In: *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003. ISSN 1532-4435.
329. HEER, J. Agency plus automation: Designing artificial intelligence into interactive systems. In: *Proc. Natl. Acad. Sci. USA*, vol. 116, no. 6, pp. 1844–1850, 2019. ISSN 0027-8424.
330. BRASOVEANU, A., ANDONIE, R. Visualizing Transformers for NLP: A Brief Survey. In: *24th International Conference Information Visualization*, 2020. ISBN 978-1-7281-9134-8.
331. PARK, D. et al. Conceptvector: Text visual analytics via interactive lexicon building using word embedding. In: *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 361–370, 2018. ISSN 1077-2626.
332. SAMEK, W., MONTAVON, G., VEDALDI, A., HANSEN, L. K., and MULLER, K., Eds., Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, ser. In: *Lecture Notes in Computer Science*. Springer, 2019, vol. 11700. ISSN 0302-9743.
333. SLACK, D. et al. How can we fool LIME and shape adversarial attacks on post hoc explanation methods. *CoRR*, vol. abs/1911.02508, 2019. ISSN 2331-8422.
334. FAN, A. et al. ELI5: long form question answering. In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28-August 2, 2019, Volume 1: Long Papers*, A. Korhonen, D. R. Traum, and L. Marquez, Eds. Association for Computational Linguistics. 2019, pp. 3558– 3567. ISBN 1950737497.
335. ZHANG, D. et al. An interpretable deep-learning model for early prediction of sepsis in the emergency department. In: *Patterns*. 2021;2:100196. ISSN 2666-3899.
336. GANDIN, I., SCAGNETTO, A., ROMANI, S., BARBATI, G. Interpretability of time-series deep learning models: a study in cardiovascular patients admitted to Intensive care unit. In: *J Biomed Inform.* 2021;121:103876. ISSN 1532-0464.
337. MORID, M.A., BORJALI, A., DEL FIOLE, G. A scoping review of transfer learning research on medical image analysis using ImageNet. In: *Comput Biol Med.* 2021;128:104115. ISSN 1879-0534.
338. TSENG, K-K. et al. Cross-domain transfer learning for pcg diagnosis algorithm. In: *Biosensors*. 2021;11:127. ISSN 2079-6374.

339. LIU, K. et al. Development and validation of a personalized model with transfer learning for acute kidney injury risk estimation using electronic health records. In: *JAMA Netw Open*. 2022;5:e2219776. ISSN 2574-3805.
340. CHEN, H. et al. Forecasting adverse surgical events using self-supervised transfer learning for physiological signals. In: *NPJ Digit Med*. 2021;4:167. ISSN 2398-6352.
341. USMAN, S.M., KHALID S., BASHIR S. A deep learning-based ensemble learning method for epileptic seizure prediction. In: *Computers in Biology and Medicine*, vol. 136, September 2021, 104710. ISSN 1879-0534.
342. IBRAHIM A.K. et al. Epileptic seizure prediction based on multiresolution convolutional neural networks. In: *Front. Sign. Processing*, vol. 3, 2023. <https://doi.org/10.2389/frisp.2023.1175395>. ISSN 0165-1684.
343. **IAPĂSCURTĂ, V.**, FIODOROV, I. NLP Tools for Epileptic Seizure Prediction Using EEG Data: A Comparative Study of Three ML Models. In: *6th International Conference on Nanotechnologies and Biomedical Engineering, September 20-23, 2023*, Springer IFMBE Proceedings 92, pp. 170–180, 2024. https://doi.org/10.1007/978-3-031-42782-4_19.
344. HONG-FEI DENG et al. Evaluating machine learning models for sepsis prediction: A systematic review of methodologies. In: *iScience* 25, 103651, January 21, 2022, pp. 1-18. ISSN 2589-0042.
345. MOONS K.G.M. et al. Risk prediction models: I. Development, internal validation, and assessing the incremental value of a new (bio)marker. In: *Heart* 2012; 98:683–90. ISSN 1355-6037.
346. COLLINS G.S. et al. External validation of multivariable prediction models: a systematic review of methodological conduct and reporting. In: *BMC Med Res Methodol* 2014; 14:40. ISSN 1471-2288.
347. STEPIEN, K. et al. Kolmogorov Complexity of Coronary Sinus Atrial Electrograms Before Ablation Predicts Termination of Atrial Fibrillation After Pulmonary Vein Isolation. In: *Entropy* 2019, 21, 970; doi:10.3390/e21100970. ISSN 1099-4300.
348. HERNÁNDEZ-ESPINOSA, A., ZENIL, H., KIANI, N.A., and TEGNÉR, J. Estimations of Integrated Information Based on Algorithmic Complexity and Dynamic Querying. In: *Handbook of Unconventional Computing*, Adamatzky, A. ed. *WSPC Book Series in Unconventional Computing*, Vol. 1 2021. https://doi.org/10.1142/9789811235726_0005. ISBN 9811235031.
349. JOHNSON, A. et al. MIMIC-IV: A comprehensive clinical database. In: *J Am Med Inform Assoc*. 2023, vol. 30(1):1-10. ISSN 1067-5027.
350. DEVLIN, J. et al. BERT: Pre-training of deep bidirectional transformers. In: *Proc NAACL*. 2019, pp. 4171-4186. ISSN 0736-587X.
351. World Health Organization and the European Centre for Disease Prevention and Control, Antimicrobial resistance surveillance. *Eurosurveillance*. 2022, 136 p. ISBN 978-92-890-5668-7.
352. KAEHLING, L.P. et al. Reinforcement learning: A survey. In: *J Artif Intell Res*. 1996, vol. 4:237-285. ISSN 1076-9757.
353. Cemri, M. et al. Why do multi-agent LLM systems fail? (Preprint arXiv:2503.13657, 2025), [cited 08.08.2025]. Available from: <https://doi.org/10.48550/arXiv.2503.13657>
354. KUMAR, A. et al. Duration of hypotension before antimicrobial therapy. In: *Crit Care Med*. 2006 vol. 34(6):1589-1596. ISSN 0090-3493.

355. OLIEHOEK, F.A. Decentralized POMDPs. In: *Wiering, M., van Otterlo, M. (eds) Reinforcement Learning. Adaptation, Learning, and Optimization*, vol. 12. 2012, Springer, Berlin, Heidelberg. ISBN 978-3-642-27645-3.
356. JOHNSON, J. et al. Billion-scale similarity search with GPUs. In: *IEEE Trans Big Data*. 2021, vol. 7(3):535-547. ISSN 2332-7790.
357. LEE, J. et al. BioBERT: A pre-trained biomedical language representation model. In: *Bioinformatics*. 2020, vol. 36(4):1234-1240. ISSN 1367-4811.
358. HOCHREITER, S., SCHMIDHUBER, J. Long short-term memory. In: *Neural Comput.* 1997, vol. 9(8):1735-1780. ISSN 0899-7667.
359. PEARL, J. *Probabilistic reasoning in intelligent systems*. 1998, Morgan Kaufmann, 584 p. ISBN 978-1558604797.
360. WATKINS, C.J.C.H., DAYAN, P. Q-learning. In: *Mach Learn.* 1992, vol. 8:279-292. ISSN 1573-0565.
361. OBERMEYER, Z. et al. Dissecting racial bias in an algorithm. In: *Science*. 2019, vol. 366(6464):447-453. ISSN 1095-9203.
362. KAMIRAN, F., CALDERS, T. Data preprocessing techniques for classification without discrimination. In: *Knowl Inf Syst.* 2012, vol. 33(1):1-33. ISSN 0219-1377.
363. SCHUETZ, P. et al. Procalcitonin to guide antibiotic therapy. In: *Lancet Infect Dis.* 2018, vol. 18(6):e143-e158. ISSN 1473-3099.
364. WU, Q. et al. AutoGen: Enabling next-gen LLM applications. (Preprint arXiv:2308.08155. 2023). [cited 08.08.2025]. Available from: <https://doi.org/10.48550/arXiv.2308.08155>
365. RUDD, K.E. et al. Global burden of sepsis. In: *Lancet*. 2020, vol. 395(10219):200-211. ISSN 0140-6736.
366. **IAPĂSCURTĂ, V., BELÎ, A.** Rolul potențial al abordării bazate pe inteligența artificială pentru managementul sepsisului: un studiu de caz pilot, În: *Arta Medica*. 2024, vol. 4 (93), pp. 46-49. ISSN 1810-1852.
367. **IAPĂSCURTĂ, V., KRONIN, S., FIODOROV, I.** (2024). Retrieval-augmented generation using domain-specific text: a pilot study. In: *Journal of Engineering Science*. 2024, vol. 31(2), pp. 48–59. ISSN 2587-3474.
368. Applications that can reason. Powered by LangChain. [online]. [cited 08.08.2025]. Available from: <https://www.langchain.com/>
369. Chroma. The AI-native open-source embedding database. [online]. [cited 08.08.2025]. Available from: <https://www.trychroma.com/>
370. LlamaIndex. [online]. [cited 08.08.2025], Available from: <https://www.llamaindex.ai/>
371. GPT-3.5 Turbo Models [online]. [cited 08.08.2025]. Available from: <https://platform.openai.com/docs/models/gpt-3-5-turbo>
372. TruEra: Building an LLM app? Evaluate, debug, and monitor. At scale. [online]. [cited 08.08.2025]. Available from: <https://truera.com/>
373. REIMERS, N., GUREVYCH, I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: *Proc EMNLP*. 2019:3982-3992. ISSN 0736-587X.
374. KONEČNÝ, J. et al. Federated learning: Strategies for improving communication efficiency. (Preprint arXiv:1610.05492. 2016). [cited 08.08.2025]. Available from: <https://doi.org/10.48550/arXiv.1610.05492>.

375. **IAPASCURTA, V.** Palmyra med sepsis app [Internet]. GitHub 2024. [online]. [cited 08.08.2025]. Available from: https://github.com/viapascurta-tech/Palmyra_med_sepsis_app
376. **IAPĂSCURTĂ, V., FIODOROV, I., BELÎL, A., BOSTAN, V.** Multi-Agent Approach for Sepsis Management. In: *Healthcare Informatics Research*. 2025, vol. 31(2), pp. 209-214. ISSN 2093-3681. <https://doi.org/10.4258/hir.2025.31.2.209>
377. The Leading Multi-Agent Platform. CrewAI [online]. [cited 08.08.2025]. Available from: <https://www.crewai.com>
378. **IAPASCUARTA, V.** Enhancing Sepsis Management in the Intensive Care Unit: A Multi-agent Methodology for Decision Support. In: *Kahraman, C., et al. Intelligent and Fuzzy Systems. INFUS 2025. Lecture Notes in Networks and Systems*, vol. 1530. Springer, Cham. pp.345-351. https://doi.org/10.1007/978-3-031-98565-2_38.
379. **CIUBARA, R., ODAJIU, O., MUNTEANU, V., ARNAUT, O., BELÎL, A., IAPASCURTA, V.** Multi-agent Decision Support for Sepsis: Balancing Precision and Hallucination Risks in Biomedical Engineering. In: *Sontea, V., Tiginyanu, I., Railean, S. (eds.) 7th International Conference on Nanotechnologies and Biomedical Engineering. ICNBME 2025. IFMBE Proceedings*, vol. 135. Springer, Cham. pp. 645-654. https://doi.org/10.1007/978-3-032-06497-4_63.
380. **IAPĂSCURTĂ, V.** *Inteligență, bazată pe complexitate în terapia intensivă: modelare, algoritmi și aplicații. Partea 1. Complexitatea în acțiune: de la modele teoretice la soluții multi-agent și gemeni digitali.* UTM 2025, 158 p. în proces
381. **IAPĂSCURTĂ, V.** *Inteligență, bazată pe complexitate în terapia intensivă: modelare, algoritmi și aplicații. Partea 2. De la date la decizii: complexitate algoritmică și învățare automată în aplicațiile de terapie intensivă.* UTM 2025, 177 p. în proces.
382. **IAPASCURTA, V.** Mother-Placenta-Fetus and Oxygen Delivery [software] (model ID 6688), [cited 12.11.2025]. Available at: https://modelingcommons.org/browse/one_model/6688.
383. **IAPĂSCURTĂ, V., ROGOVSCHI, N., GROZAVU, N., FIODOROV, I.** Kolmogorov-Chaitin algorithmic complexity and text analysis through the prism of multiview approach for EEG analysis. In: *Numerical Analysis and Scientific Computation with Applications, NASCA23*, 3-6 July, 2023, Athens, Greece https://www.math.uoa.gr/anakoinoseis_kai_ekdiloseis/proboli_anakoinosis/3_6_ioylioy_2023_numerical_analysis_and_scientific_computation_with_applications_nas
384. **IAPĂSCURTĂ, V.** Advancing Critical Care with Digital Twins: IntegratingSystem Dynamics Modeling of the Circulatory System. *Congress SARRM 2025*. [cited 7.10.2026]. Available at: <https://repository.usmf.md/handle/20.500.12710/33705>
385. **IAPĂSCURTĂ, V.** Unveiling Large Language Model Vulnerabilities in Intensive Care: Insights from Sepsis Management Applications. *Congress SARRM 2025*. [cited 7.10.2026]. Available at: <https://repository.usmf.md/handle/20.500.12710/33706>
386. **IAPĂSCURTĂ, V.** Agent-based modeling of fluid dynamics in lung tissue engineering. In: *The Materials of the National Scientific Conference with International Participation „Cells and tissues transplantation. Actualities and perspectives. The 3rd*

- edition" dedicated to the 80th anniversary of the founding of Nicolae Testemitanu State University of Medicine and Pharmacy. Chisinau, March 21-22, 2025. ISBN 978-9975-82-413-2. <http://repository.usmf.md/handle/20.500.12710/30460>
387. **IAPĂSCURTĂ, V.** Building a RAG system for tissue engineering: insights from domain specific text and sepsis management, In: *The Materials of the National Scientific Conference with International Participation „Cells and tissues transplantation. Actualities and perspectives. The 3rd edition" dedicated to the 80th anniversary of the founding of Nicolae Testemitanu State University of Medicine and Pharmacy.* Chisinau, March 21-22, 2025. ISBN 978-9975-82-413-2. <http://repository.usmf.md/handle/20.500.12710/30459>
 388. **IAPĂSCURTĂ, V.** Modeling the Human Circulatory System Using System Dynamics: Possibilities, Benefits, and Practical Applications. In: *International Conference on Electronics, Communications and Computing: IC ECCO 2024, Ed. 13, 17-18 octombrie 2024, Chişinău.* Chişinău: „Tehnica-UTM," 2024, Editia 13, pp. 43-44. ISBN (pdf) 978-9975-64-480-8 (PDF). https://ibn.idsi.md/en/vizualizare_articol/223087.
 389. **IAPĂSCURTĂ, V., TURCANU, D., SIMINIUC, R.** Integration of a Proprietary Software Application and a Multimodal LLM for Enhanced Nutritional Guidance. In: *International Conference on Electronics, Communications and Computing: IC ECCO 2024, Ed. 13, 17-18 octombrie 2024, Chişinău.* Chişinău: „Tehnica-UTM," 2024, Editia 13, p. 156. ISBN 978-9975-64-480-8 (PDF). https://ibn.idsi.md/en/vizualizare_articol/223317.
 390. **IAPĂSCURTĂ, V., BELÎI, A.** Combinarea modelelor de predicție a sepsisului cu modul RAG integrat: îmbunătățirea suportului decizional clinic în UTI. In: *Cercetarea în biomedicină și sănătate: calitate, excelență și performanță, Ed. 4, 16-18 octombrie 2024, Chişinău.* Chişinău, Republica Moldova: 2024, p. 559. ISSN 2345-1467. https://ibn.idsi.md/en/vizualizare_articol/224580.
 391. **IAPĂSCURTĂ, V., BOLEAC, D., FIODOROV, I.** Introducerea unei metode de pionierat bazate pe EEG pentru monitorizarea stării creierului în îngrijirea neurointensivă, In: *Cercetarea în biomedicină și sănătate: calitate, excelență și performanță, Ed. 4, 16-18 octombrie 2024, Chişinău.* Chişinău, Republica Moldova: 2024, p. 564. ISSN 2345-1467. https://ibn.idsi.md/en/vizualizare_articol/224642.
 392. **IAPĂSCURTĂ, V., LISNIC, V., FIODOROV, I.** O nouă abordare a analizei EEG bazată pe complexitatea algoritmică cu potențial de adnotare a crizelor epileptice: dovada conceptului, In: *Cercetarea în biomedicină și sănătate: calitate, excelență și performanță, Ed. 4, 16-18 octombrie 2024, Chişinău.* Chişinău, Republica Moldova: 2024, p. 402. ISSN 2345-1467. https://ibn.idsi.md/en/vizualizare_articol/223876.
 393. **IAPĂSCURTĂ, V.** Agent-based modeling: revolutionizing tissue engineering. In: *The Materials of the National Scientific Conference with International Participation, the 2nd edition, Chisinau, March 29-30th, 2024.* ISBN 978-9975-82-366-1 <https://repository.usmf.md/handle/20.500.12710/27016>.
 394. **IAPĂSCURTĂ, V.** The Combined Use of Agent-Based Modeling (ABM) and System Dynamics Modeling (SDM) for Tissue Engineering: A Raw Example of Interaction at Different Scales, In: *The Materials of the National Scientific Conference with*

International Participation, the 2nd edition, Chisinau, March 29-30th, 2024. ISBN 978-9975-82-366-1. <https://repository.usmf.md/handle/20.500.12710/27015>.

395. **IAPĂSCURTĂ, V., FIODOROV, I.** Inteligența artificială pentru prezicerea stării clinice a pacientului pe exemplul sepsisului. In: *Conferința științifică anuală a USMF „N. Testemițanu.” Cercetarea în biomedicină și sănătate: calitate, excelență și performanță, 18-20 octombrie, 2023, Chișinău.* Moldovan Journal of Health Sciences, ISSN 2345-1467. <https://repository.usmf.md/handle/20.500.12710/25611>.
396. **IAPASCURTĂ V., FIODOROV, I.** Kolmogorov-Chaitin Algorithmic Complexity for EEG Analysis. In: *Electronics, Communications and Computing, Ed. 12, 20–21, October 2022, Chișinău.* Chișinău: Tehnica-UTM, 2023, Editia 12, pp. 214-218. DOI: 10.52326/ic-ecco.2022/CS. ISBN 978-9975-45-898-6. <https://doi.org/10.52326/ic-ecco.2022/CS.14>
397. **IAPĂSCURTĂ, V.** Detection of Movement toward Randomness by Applying the Block Decomposition Method to a Simple Model of the Circulatory System. In: *Complex Systems Journal.* 2019, 28(1), pp. 59-77. ISSN 0891-2513. pp. <https://doi.org/10.25088/ComplexSystems.28.1.59>
398. **IAPĂSCURTĂ, V., ȚURCANU, D., BELÎI, A., BOSTAN, V.** Multi-Agent System for Early Sepsis Management Support: A Follow-up Evaluation Study. In: *Healthcare Informatics Research.* 2026, vol. 32(2), pp. 180-195. ISSN 2093-3681. <https://doi.org/10.4258/hir.2026.32.2.190>
399. **IAPASCUARTA, V.** Tracking the Dynamics of a Simulated Myocardial Infarction in a System Dynamics Model of the Human Circulatory System Using Kolmogorov-Chaitin Complexity. In: *Kahraman, C., et al. Intelligent and Fuzzy Systems. INFUS 2026. Lecture Notes in Networks and Systems,* Springer, Cham. forthcoming
400. **IAPĂSCURTĂ, V., ȚURCANU, D., BELÎI, A., BOSTAN, V.** Tracking the temporal dynamics of a system dynamics model of the human circulatory system with Kolmogorov-Chaitin complexity: toward digital twins for the intensive care unit. In: *Journal of Engineering Science.* 2026. ISSN 2587-3474. forthcoming.
401. **IAPĂSCURTĂ, V., ȚURCANU, D., BELÎI, A., BOSTAN, V.** Halucinațiile modelelor de limbaj mari în sepsis: sinteză narativă și analiza a zece cazuri. In: *Arta Medica,* 2026. forthcoming.

SUPPLEMENTARY MATERIALS

Appendix 1

Dr. Hector Zenil
Associate Professor
School of Biomedical Engineering and Imaging Sciences
King's Institute for Artificial Intelligence



March 4, 2025

To Whom It May Concern

Subject: Confirmation of Victor Iapascorta's Participation in Algorithmic Information Dynamics Research Project

I am writing to officially confirm the active and pivotal role of Dr. Victor Iapascorta, PhD, within the Algorithmic Information Dynamics research project. Dr. Iapascorta serves as the coordinator of the subgroup dedicated to AID & (Patho)physiological and Clinical-end Models, contributing significantly to the innovative research efforts of the Algorithmic Information Dynamics research project.

The Algorithmic Information Dynamics research project stands as a distinguished private initiative that unites researchers from various esteemed institutions, fostering a collaborative environment for cutting-edge exploration. At its core lies a multidisciplinary team, with algorithmic information dynamics serving as the cornerstone of our collective research endeavors.

Dr. Iapascorta's leadership of the AID & (Patho)physiological and Clinical-end Models team showcases his expertise in applying algorithmic information dynamics to the realms of biology and medicine, enriching our project with valuable insights and advancements.

Commencing in 2018, the project continues to evolve and flourish with each passing day. Our team comprises select members who have demonstrated exceptional aptitude, including Dr. Iapascorta, who was among the distinguished participants of the course "Algorithmic Information Dynamics: A Computational Approach to Causality and Living Systems From Networks to Cells" held at the Santa Fe Institute/Complexity Explorer in 2018.

I appreciate your attention to this matter.

With warm regards,

A handwritten signature in black ink, appearing to be "Hector Zenil", written over a faint grid pattern.

Dr. Hector Zenil

Appendix 2

Non-linear regression methods for missing values imputation

We experimented with a number of nonlinear methods, including exponential regression and polynomial regression of varying degrees, along with linear methods to determine the most appropriate method for imputing missing values in columns with predominantly missing values. Among nonlinear methods, quadratic regression had the best fit to the study data, while exponential regression was inferior. The results of these experiments are presented in this appendix.

a) Quadratic Regression

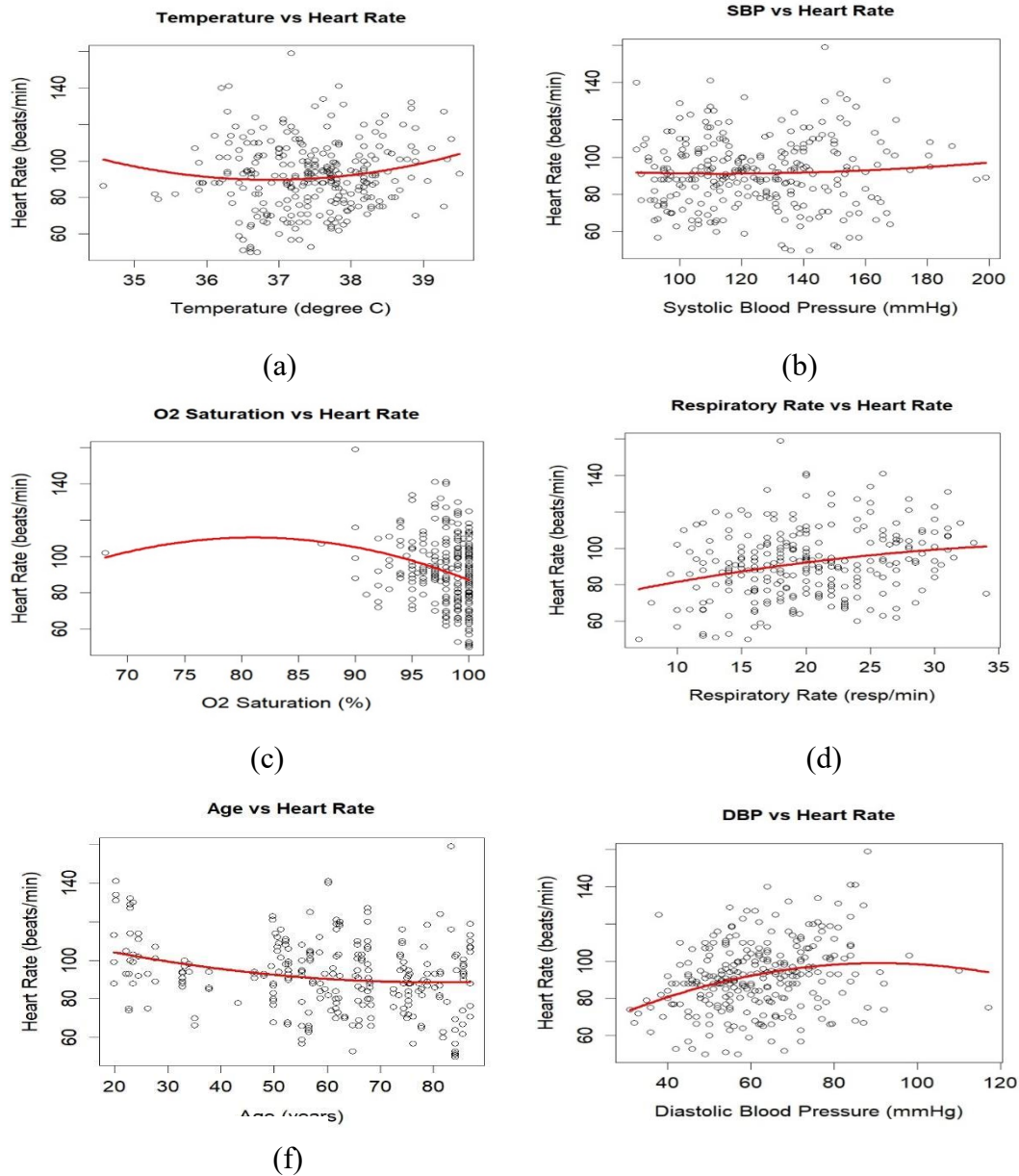


Figure A2.1. Quadratic regression involving the heart rate versus other variables: (a) temperature; (b) systolic blood pressure; (c) O₂ saturation; (d) respiratory rate; (e) age and (f) diastolic blood pressure. Red lines represent the best fits.

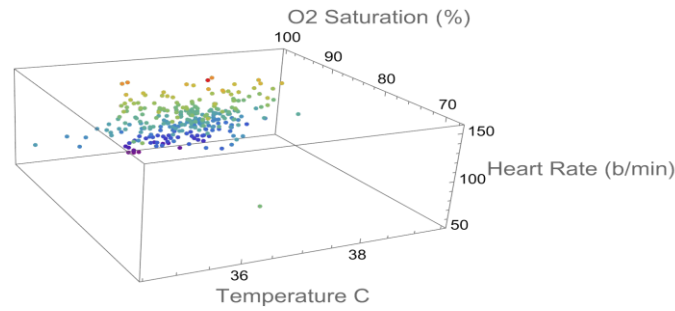


Figure A2.2. 3D plot for three variables: O₂ saturation, heart rate, and body temperature.

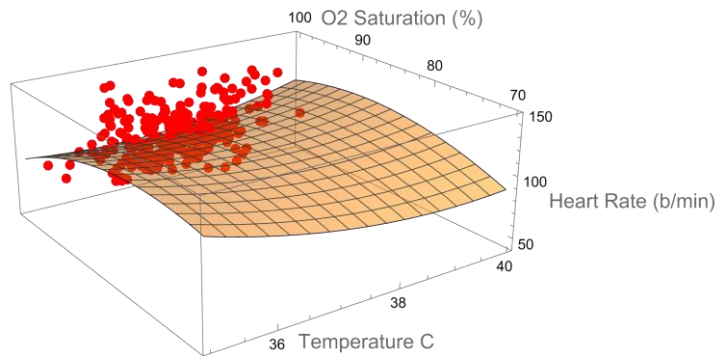


Figure A2.3. 3D plot exemplifying the quadratic regression approach to the data in Figure A6.2.

Equation A2.1 depicts the quadratic model that describes the relationship between the data presented in Figures A2.2 and A2.3.

$$\begin{aligned} \text{Heart Rate} = & -854.611 + 8.2499 \times \text{Temperature} + 0.6683 \times \text{Temperature}^2 \quad (\text{A2.1}) \\ & + 15.3291 \times \text{O}_2\text{Saturation} - 0.5505 \times \text{Temperature} \times \text{O}_2\text{Saturation} \\ & + 0.0224 \times \text{O}_2\text{Saturation}^2 \end{aligned}$$

b) Exponential regression

Compared with polynomial regression, the results provided by exponential regression proved to be inferior while analyzing the sepsis data set. Figure A2.4 and Equation A2.2 exemplify an exponential model with three variables: heart rate, body temperature, and systolic blood pressure.

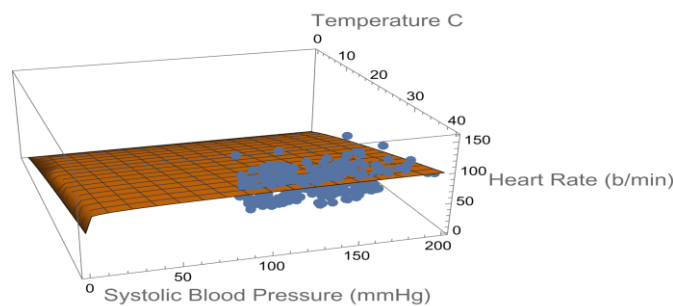


Figure A2.4. 3D plot exemplifying an exponential regression model for three variables: temperature, heart rate, and systolic blood pressure.

The equation A2.2 depicts the exponential model built on these data.

$$\text{Heart Rate} = 3.6162 \times \text{Temperature}^{0.9561} \times \text{Systolic Blood Pressure}^{0.0271} \quad (\text{A2.2})$$

Above are presented examples of quadratic and exponential regression. The experiments also included higher-than-degree-two polynomial regression modeling. Figure A2.5 shows this approach. It depicts a visual representation of polynomial models from degree 2 to degree 5 for relationships concerning heart rate and oxygen saturation.

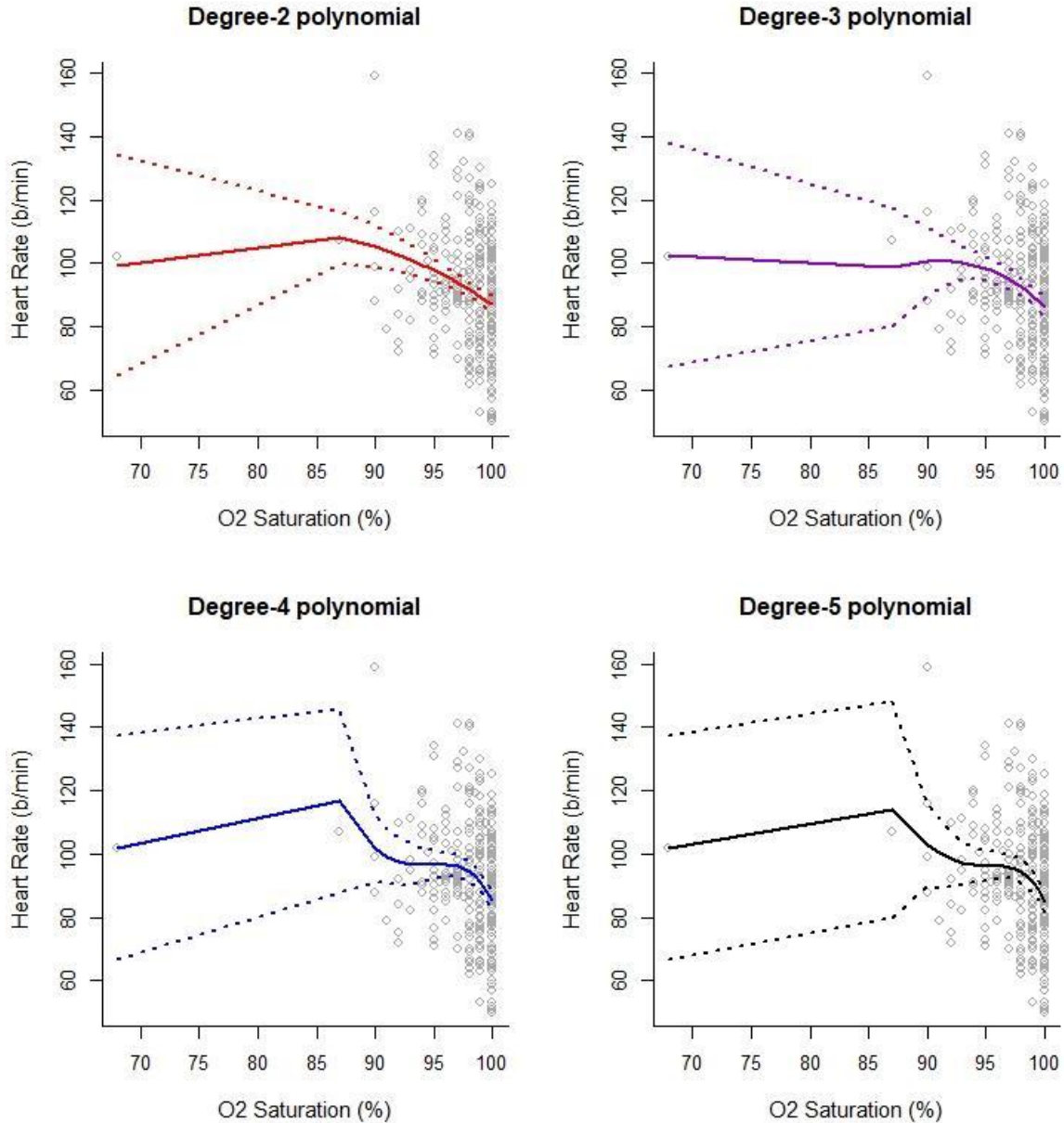


Figure A2.5. Polynomial models from degree 2 to degree 5 for relationships concerning heart rate and oxygen saturation. The dotted lines on either side of the solid-line fitted curve are $2 \times$ standard error curves.

With a simple visual inspection, it is difficult to select the model with the best fit. To base the decision on a more formal procedure, the classical statistical methodology of analysis-of-variance (ANOVA) was used. A sequential comparison based on the F-test was performed, comparing first the linear model vs. the quadratic model (degree-2 polynomial), then the quadratic

model vs. the cubic model (degree-3 polynomial), and so on. The ANOVA results indicated that the quadratic model was the best fit for this particular data set with a statistical significance level from 0.1 to 0.05 for three out of six variables (i.e., temperature, O₂ saturation, and diastolic blood pressure).

The data for these experiments with non-linear approaches come from septic patients and represent complete cases, similar to the experiments with linear models. Extracted complete cases were split into a training and test set. The figures above are based on data from the test set. As for the models for missing data imputation, the training set is used, and the equation representations for the best quadratic models are depicted by the equations A2.3 to A2.8.

$$\begin{aligned} \text{Heart rate} = & 64.2967 + 0.3928 \times \text{diastolic blood pressure} \\ & - 0.0025 \times \text{diastolic blood pressure}^2 - 0.1454 \times \text{respiratory rate} \\ & + 0.0112 \times \text{diastolic blood pressure} \times \text{respiratory rate} \\ & - 0.0018 \times \text{respiratory rate}^2 \end{aligned} \quad (\text{A2.3})$$

$$\begin{aligned} \text{O}_2 \text{ saturation} \\ = & 99.8990 + 0.0264 \times \text{heart rate} - 0.0004 \times \text{heart rate}^2 \\ & - 0.1823 \times \text{respiratory rate} + 0.0015 \times \text{heart rate} \times \text{respiratory rate} \\ & - 0.0015 \times \text{respiratory rate}^2 \end{aligned} \quad (\text{A2.4})$$

$$\begin{aligned} \text{Temperature} = & 34.6761 + 0.0406 \times \text{heart rate} - 0.0001 \times \text{heart rate}^2 \\ & + 0.0163 \times \text{age} - 0.0002 \times \text{heart rate} \times \text{age} - 0.00002 \times \text{age}^2 \end{aligned} \quad (\text{A2.5})$$

$$\begin{aligned} \text{Systolic blood pressure} \\ = & -167.0507 + 5.7565 \times \text{temperature} \\ & + 0.0199 \times \text{temperature}^2 + 3.1869 \times \text{diastolic blood pressure} \\ & - 0.0689 \times \text{temperature} \times \text{diastolic blood pressure} \\ & + 0.0025 \times \text{diastolic blood pressure}^2 \end{aligned} \quad (\text{A2.6})$$

$$\begin{aligned} \text{Diastolic blood pressure} \\ = & 33.7832 + 0.2306 \times \text{systolic blood pressure} \\ & + 0.0008 \times \text{systolic blood pressure}^2 + 0.1198 \times \text{age} \\ & - 0.0024 \times \text{systolic blood pressure} \times \text{age} - 0.0007 \times \text{age}^2 \end{aligned} \quad (\text{A2.7})$$

$$\begin{aligned} \text{Respiratory rate} \\ = & -25.8640 - 0.6420 \times \text{heart rate} + 0.0002 \times \text{heart rate}^2 \\ & + 2.1470 \times \text{O}_2 \text{ saturation} + 0.0067 \times \text{heart rate} \times \text{O}_2 \text{ saturation} \\ & - 0.0174 \times \text{O}_2 \text{ saturation}^2 \end{aligned} \quad (\text{A2.8})$$

We used these models for missing value imputation, substituting the values in a specific column of the training set with NA. Then the generated data in every column were compared with the original data, and the statistics for this comparison were estimated (e.g., Pearson's correlation coefficient, paired t-test, and Wilcoxon signed-rank/paired test). The results are presented in Table A2.1.

The same procedure was performed with the use of GLM (described in the main text of the dissertation). Table A2.2 shows the statistics for filling in the missing values for all six variables (heart rate, O₂Sat, systolic blood pressure, diastolic blood pressure, temperature, respiratory rate) using GLM.

Table A2.1. Statistics for the recovery of the missing values with the use of Quadratic Regression Models

Statistics\ Recovered variable	Heart Rate (beats/min)	O ₂ Sat (%)	Temperature (degree C)	Systolic Blood Pressure (mmHg)	Diastolic Blood Pressure (mmHg)	Respiratory Rate (resp/min)
Pearson's correlation coefficient	0.3743679	0.274203	0.2616435	0.6021594	0.6719851	0.3162864
Paired t-test p-value	0.007672601	0.2608096	0.00001659	0.2360559	0.8531508	0.5821566
Wilcoxon signed-rank test p-value	0.007625574	0.0006602	0.00000854	0.675608	0.3214859	0.9081279

Table A2.2. Statistics for the recovery of the missing values with the use of GLM

Statistics\ Recovered variable	Heart Rate (beats/min)	O ₂ Sat (%)	Temperature (degree C)	Systolic Blood Pressure (mmHg)	Diastolic Blood Pressure (mmHg)	Respiratory Rate (resp/min)
Pearson's correlation coefficient	0.3827774	0.3077319	0.2970013	0.6075263	0.7052836	0.3413348
Paired t-test p-value	0.03141058	0.0949303	0.0002421	0.5854771	0.2142654	0.1365223
Wilcoxon signed-rank test p-value	0.03674085	0.00004062	0.0001375	0.9040381	0.04345416	0.07573581

As can be seen from the tables above, the statistics for both methods are similar, with a slightly higher correlation coefficient in the case of data generated by GLM. Since some variables are not normally distributed (e.g., O₂ saturation), the role of the student's t-test is limited, and the paired Wilcoxon test seems more appropriate. When using the Wilcoxon test, if the p-value is less than or equal to the significance level (for example, 0.05), the results are statistically significant and the null hypothesis (i.e., the median difference between pairs is zero) should be rejected, which means that with this perspective, the generated data differs from the original data. Based on these results and lower computational costs, the GLM approach was chosen for further data processing.

Appendix 3

General description of the architecture of a BERT model built in the study and BERT classifier

Dog_1 Bert model architecture

The BERT model has 105 different named parameters.

==== Embedding Layer ====

```
bert.embeddings.word_embeddings.weight      (734, 768)
bert.embeddings.position_embeddings.weight  (514, 768)
bert.embeddings.token_type_embeddings.weight (1, 768)
bert.embeddings.LayerNorm.weight           (768,)
bert.embeddings.LayerNorm.bias             (768,)
```

==== First Transformer ====

```
bert.encoder.layer.0.attention.self.query.weight (768, 768)
bert.encoder.layer.0.attention.self.query.bias  (768,)
bert.encoder.layer.0.attention.self.key.weight  (768, 768)
bert.encoder.layer.0.attention.self.key.bias    (768,)
bert.encoder.layer.0.attention.self.value.weight (768, 768)
bert.encoder.layer.0.attention.self.value.bias  (768,)
bert.encoder.layer.0.attention.output.dense.weight (768, 768)
bert.encoder.layer.0.attention.output.dense.bias (768,)
bert.encoder.layer.0.attention.output.LayerNorm.weight (768,)
bert.encoder.layer.0.attention.output.LayerNorm.bias (768,)
bert.encoder.layer.0.intermediate.dense.weight (3072, 768)
bert.encoder.layer.0.intermediate.dense.bias   (3072,)
bert.encoder.layer.0.output.dense.weight       (768, 3072)
bert.encoder.layer.0.output.dense.bias         (768,)
bert.encoder.layer.0.output.LayerNorm.weight   (768,)
bert.encoder.layer.0.output.LayerNorm.bias     (768,)
```

==== Output Layer ====

```
bert.pooler.dense.weight (768, 768)
bert.pooler.dense.bias   (768,)
classifier.weight         (2, 768)
classifier.bias           (2,)
```

Dog_2 BERT Classifier*

```
BertForSequenceClassification(
  (bert): BertModel(
    (embeddings): BertEmbeddings(
      (word_embeddings): Embedding(803, 768, padding_idx=0)
      (position_embeddings): Embedding(514, 768)
      (token_type_embeddings): Embedding(1, 768)
      (LayerNorm): LayerNorm((768,), eps=1e-05,
        elementwise_affine=True)
      (dropout): Dropout(p=0.1, inplace=False)
    )
    (encoder): BertEncoder(
      (layer): ModuleList(
        (0): BertLayer(
          (attention): BertAttention(
```

```

(self): BertSelfAttention(
  (query): Linear(in_features=768, out_features=768, bias=True)
  (key): Linear(in_features=768, out_features=768, bias=True)
  (value): Linear(in_features=768, out_features=768, bias=True)
  (dropout): Dropout(p=0.1, inplace=False)
)
(output): BertSelfOutput(
  (dense): Linear(in_features=768, out_features=768, bias=True)
  (LayerNorm): LayerNorm((768,), eps=1e-05,
    elementwise_affine=True) (dropout): Dropout(p=0.1,
    inplace=False)
)
)
(intermediate): BertIntermediate(
  (dense): Linear(in_features=768, out_features=3072, bias=True)
  (intermediate_act_fn): GELUActivation()
)
(output): BertOutput(
  (dense): Linear(in_features=3072, out_features=768, bias=True)
  (LayerNorm): LayerNorm((768,), eps=1e-05,
    elementwise_affine=True) (dropout): Dropout(p=0.1,
    inplace=False)
)
)
(1): BertLayer(
  (attention): BertAttention(
    (self): BertSelfAttention(
      (query): Linear(in_features=768, out_features=768, bias=True)
      (key): Linear(in_features=768, out_features=768, bias=True)
      (value): Linear(in_features=768, out_features=768, bias=True)
      (dropout): Dropout(p=0.1, inplace=False)
    )
    (output): BertSelfOutput(
      (dense): Linear(in_features=768, out_features=768, bias=True)
      (LayerNorm): LayerNorm((768,), eps=1e-05,
        elementwise_affine=True) (dropout): Dropout(p=0.1,
        inplace=False)
    )
  )
  (intermediate): BertIntermediate(
    (dense): Linear(in_features=768, out_features=3072, bias=True)
    (intermediate_act_fn): GELUActivation()
  )
  (output): BertOutput(
    (dense): Linear(in_features=3072, out_features=768, bias=True)
    (LayerNorm): LayerNorm((768,), eps=1e-05,
      elementwise_affine=True) (dropout): Dropout(p=0.1,
      inplace=False)
  )
)
)
(2): BertLayer(
  (attention): BertAttention(
    (self): BertSelfAttention(
      (query): Linear(in_features=768, out_features=768, bias=True)
      (key): Linear(in_features=768, out_features=768, bias=True)
      (value): Linear(in_features=768, out_features=768, bias=True)
      (dropout): Dropout(p=0.1, inplace=False)
    )
    (output): BertSelfOutput(

```

```

(dense): Linear(in_features=768, out_features=768, bias=True)
(LayerNorm): LayerNorm((768,), eps=1e-05,
elementwise_affine=True) (dropout): Dropout(p=0.1,
inplace=False)
)
)
(intermediate): BertIntermediate(
(dense): Linear(in_features=768, out_features=3072, bias=True)
(intermediate_act_fn): GELUActivation()
)
(output): BertOutput(
(dense): Linear(in_features=3072, out_features=768, bias=True)
(LayerNorm): LayerNorm((768,), eps=1e-05,
elementwise_affine=True) (dropout): Dropout(p=0.1,
inplace=False)
)
)
(3): BertLayer(
(attention): BertAttention(
(self): BertSelfAttention(
(query): Linear(in_features=768, out_features=768, bias=True)
(key): Linear(in_features=768, out_features=768, bias=True)
(value): Linear(in_features=768, out_features=768, bias=True)
(dropout): Dropout(p=0.1, inplace=False)
)
(output): BertSelfOutput(
(dense): Linear(in_features=768, out_features=768, bias=True)
(LayerNorm): LayerNorm((768,), eps=1e-05,
elementwise_affine=True) (dropout): Dropout(p=0.1,
inplace=False)
)
)
(intermediate): BertIntermediate(
(dense): Linear(in_features=768, out_features=3072, bias=True)
(intermediate_act_fn): GELUActivation()
)
(output): BertOutput(
(dense): Linear(in_features=3072, out_features=768, bias=True)
(LayerNorm): LayerNorm((768,), eps=1e-05,
elementwise_affine=True) (dropout): Dropout(p=0.1,
inplace=False)
)
)
(4): BertLayer(
(attention): BertAttention(
(self): BertSelfAttention(
(query): Linear(in_features=768, out_features=768, bias=True)
(key): Linear(in_features=768, out_features=768, bias=True)
(value): Linear(in_features=768, out_features=768, bias=True)
(dropout): Dropout(p=0.1, inplace=False)
)
(output): BertSelfOutput(
(dense): Linear(in_features=768, out_features=768, bias=True)
(LayerNorm): LayerNorm((768,), eps=1e-05,
elementwise_affine=True) (dropout): Dropout(p=0.1,
inplace=False)
)
)
(intermediate): BertIntermediate(

```

```

(dense): Linear(in_features=768, out_features=3072, bias=True)
(intermediate_act_fn): GELUActivation()
)
(output): BertOutput(
(dense): Linear(in_features=3072, out_features=768, bias=True)
(LayerNorm): LayerNorm((768,), eps=1e-05,
elementwise_affine=True) (dropout): Dropout(p=0.1,
inplace=False) )
)
(5): BertLayer(
(attention): BertAttention(
(self): BertSelfAttention(
(query): Linear(in_features=768, out_features=768, bias=True)
(key): Linear(in_features=768, out_features=768, bias=True)
(value): Linear(in_features=768, out_features=768, bias=True)
(dropout): Dropout(p=0.1, inplace=False)
)
(output): BertSelfOutput(
(dense): Linear(in_features=768, out_features=768, bias=True)
(LayerNorm): LayerNorm((768,), eps=1e-05,
elementwise_affine=True) (dropout): Dropout(p=0.1,
inplace=False)
)
)
(intermediate): BertIntermediate(
(dense): Linear(in_features=768, out_features=3072, bias=True)
(intermediate_act_fn): GELUActivation()
)
(output): BertOutput(
(dense): Linear(in_features=3072, out_features=768, bias=True)
(LayerNorm): LayerNorm((768,), eps=1e-05,
elementwise_affine=True) (dropout): Dropout(p=0.1,
inplace=False)
)
)
)
(pooler): BertPooler(
(dense): Linear(in_features=768, out_features=768, bias=True)
(activation): Tanh()
)
)
(dropout): Dropout(p=0.1, inplace=False)
(classifier): Linear(in_features=768, out_features=2,
bias=True)
)

```

* some indentation is ignored in order to fit the page width

Appendix 4

Graphica user interface (GUI) and code for additional software applications built in the study

A. GUI of the additional software application for EEG analysis and sepsis prediction
I. EEG Analysis (R/Shiny)

Algorithmic Complexity (by BDM) EEG Analyser

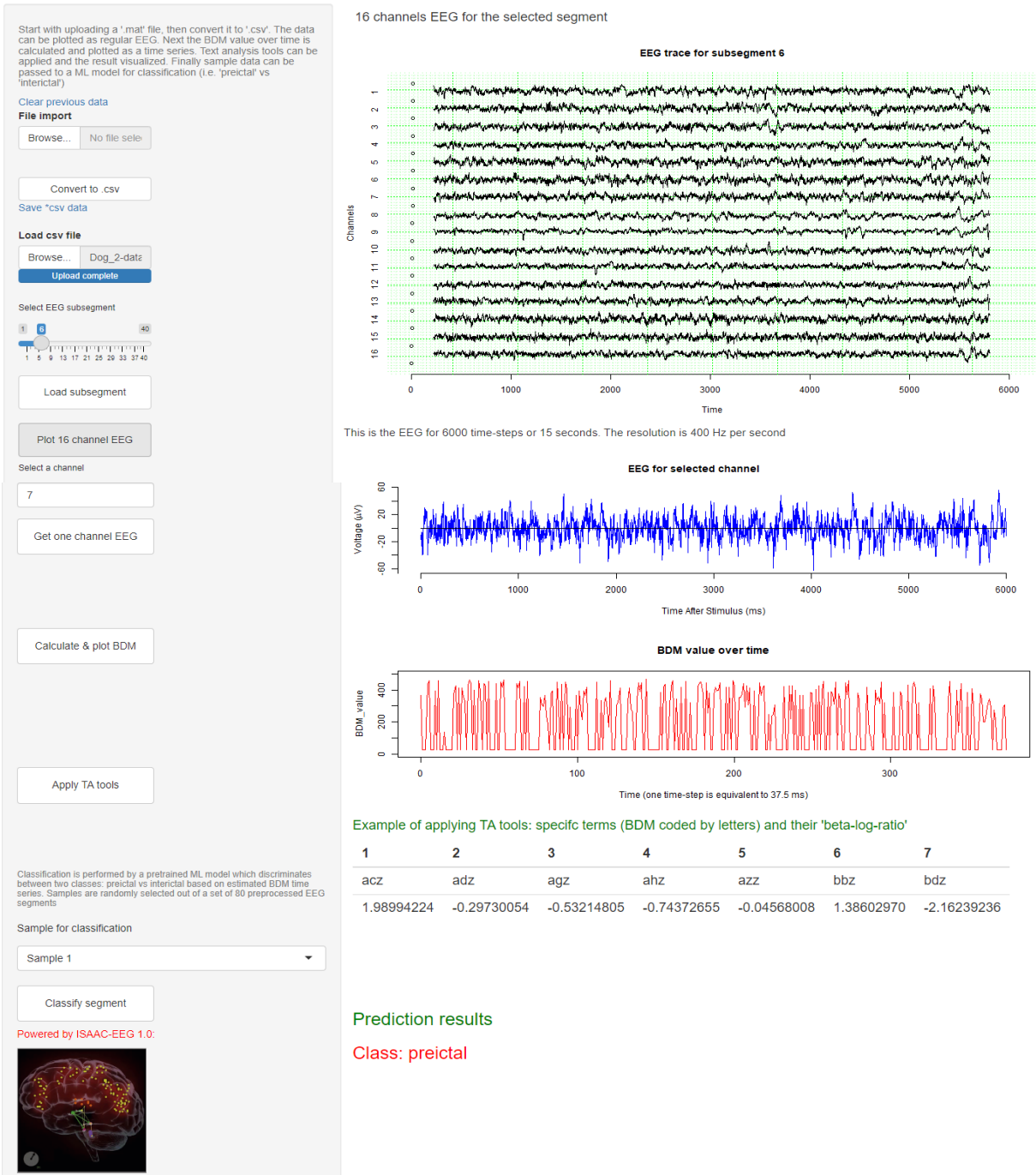


Figure A4.1. Graphical interface of the software application for EEG analysis based on Kolmogorov-Chaitin complexity and text analysis (TF-IDF approach).

II. Software application for sepsis prediction

a) Demo version of the software application for sepsis prediction (R/Shiny)

Sepsis Prediction System: ISAAC-sepsis-1.0.1 (demo)

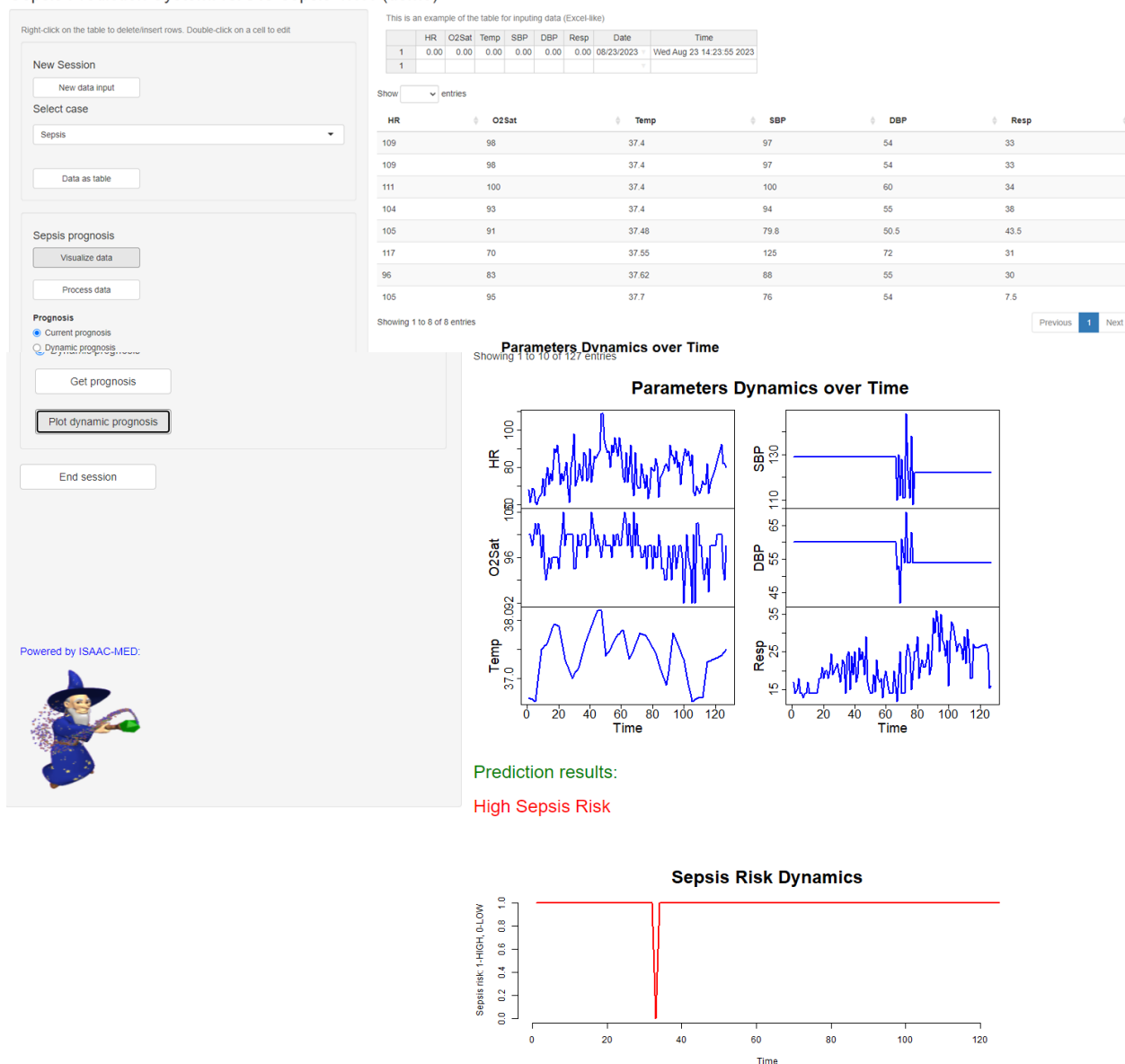


Figure A4.2. Graphical interface of the demo version of the software application for sepsis prediction

b) Python Wave h2o software application for sepsis prediction (desktop version)

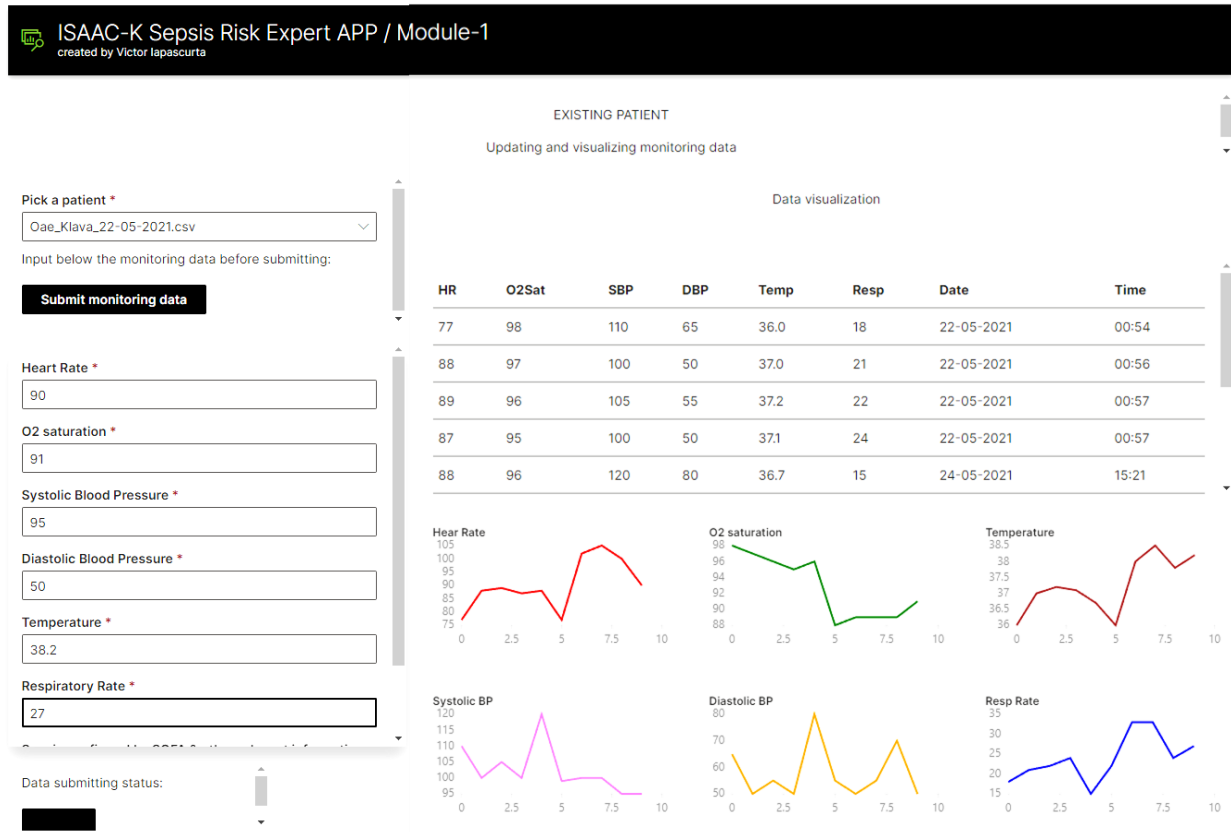


Figure A4.3. Graphical interface of the software application for sepsis prediction: entering data and viewing it in tabular and graphical form.

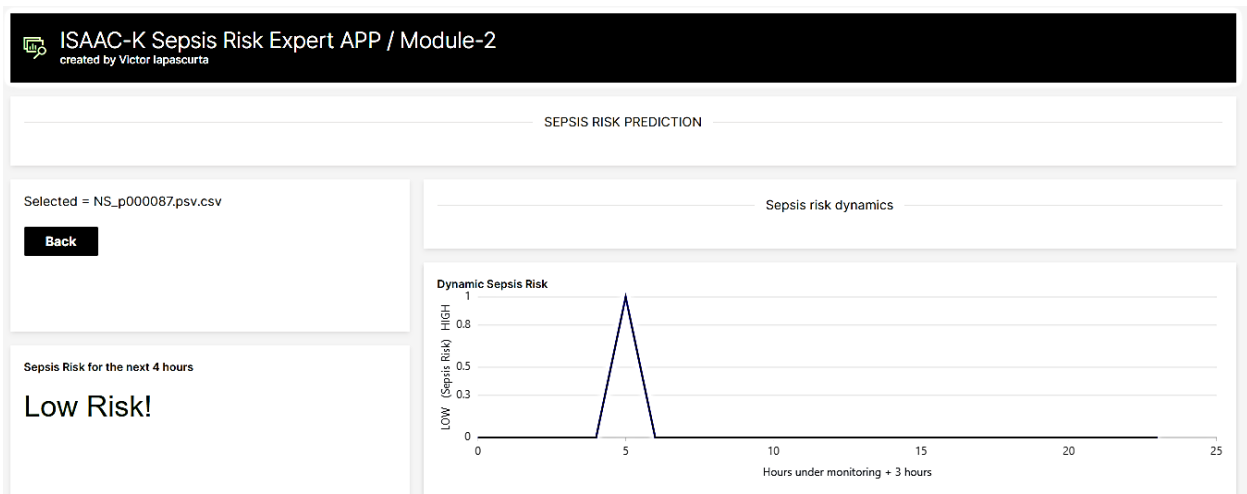


Figure A4.4. Graphical interface of the sepsis prediction application: obtaining the result of the prediction and visualizing the dynamics of sepsis risk.


ISAAC-K Sepsis Risk Expert APP
/ Module-3

NEW PATIENT

input patient information

Patient surname *

Patient first name *

Date of birth

Gender:

☒ Male
 ☐ Female

Heart Rate *

O2 saturation *

Systolic Blood Pressure *

Diastolic Blood Pressure *

Temperature *

Respiratory Rate *

Diagnosis & other relevant information

Verify

Submit data

Figure A4.5. Graphical interface of the software application for sepsis prediction: entering the data of a new patient.

B. The code for selected software applications

- **EEG analysis (KCC & NLP/Word2vec, R/Shiny)**

User interface module:

```
library(shiny)
library(shinybusy)

options(shiny.maxRequestSize = 30*1024^2)

shinyUI(fluidPage(
  add_busy_spinner(spin = "fading-circle"),
  titlePanel("Algorithmic Complexity (by BDM & NLP/Word2Vec model)
            EEG Analyser"),

  sidebarLayout(
    sidebarPanel(width=3,
      h5(
        helpText
        ("Start with uploading a '.mat' file, then convert it to
        '.csv'. The data can be plotted as regular EEG.Next the
        BDM value over time is calculated and plotted as a time
        series. Text analysis tools can be applied and the
        result visualized. Finally sample data can be passed to
        a ML model for classification (i.e. 'preictal' vs
        'interictal')")),

    actionLink("reset", "Clear previous data"),
```

[illegible]

```

actionButton("action4", label = h5("WordCloud"), width=180),

br(),
br(),
br(),
br(),

        actionButton("action5", label = h5("WordFrquencies"),
            width=180),

br(),
br(),
br(),
br(),
br(),
br(),
br(),
br(),
br(),
br(),

        actionButton("action6", label = h5("WordVectors"),
            width=180),

br(),
br(),
br(),
br(),
br(),
br(),
br(),
br(),
br(),
br(),

h6("Classification is performed by a pretrained W2v/GBM
    model which discriminates between two classes: preictal
    vs interictal based on estimated BDM time series.
    Samples are randomly selected out of a set of 80
    preprocessed EEG segments",
    style ="color:grey"),

selectInput("select", label = h5("Sample for
    cassification"), choices = list("Sample 1" = 1,
    "Sample 2" = 2, "Sample 3" = 3,"Sample 4" = 4,
    "Sample 5" = 5), selected = 1),

verbatimTextOutput("value"),

actionButton("action7", label = h5("Classify segment"),
    width=180),

h5("Powered by ISAAC-EEG 1.0:",style ="color:red"),
tags$img(src="app01.gif", contentType="image/gif",
    height=170, width=170,nonce=runif(1))

),

```

```

mainPanel(
  fluidRow(
    tabsetPanel(
      id = "conditionedPanels",

      tabPanel(value=1,
        h4('Expert'),

        h4("16 channels EEG for the selected segment"),

        plotOutput("EEG", height = 570, width = 970),
        div(p("This is the EEG for 6000 time-steps or 15 seconds. The
          resolution is 400 Hz per second"), style = "font-size: 110%"),

        plotOutput("Chan_EEG", height = 250, width = 920),

        plotOutput("BDM_TS", height = 250, width = 920),

        fluidRow(
          column(4,plotOutput('WORD_CLOUD')),
          column(5,plotOutput('WORD_FR'))
        ),

        h4("First 25 word vectors for 10 most frequent words in this
          subsegment", style="color:blue"),

        h4(verbatimTextOutput("Result0")),

        br(),
        br(),
        br(),
        br(),

        h3("Prediction results", style="color:blue"),
        h3(textOutput("Result"), style="color:red")

      ),

      tabPanel(value=2,
        h4('How to use this application'),

        hr(), h4(
          tags$div(
            "This software application exemplifies how algorithmic
            complexity and natural language processing domain
            knowledge can be used for a practical purpose: predicting
            if an EEG clip is a normal (interictal) or a pre-
            epileptic (preictal) one.",
            tags$br(),
            tags$br(),
            "Original data come as '.mat' files with EEG recordings
            from a Kaggle competition (American Epilepsy Society
            Seizure Prediction Challenge, 2014,
            www.kaggle.com/c/seizure), where sample files can be
            downloaded from.",
            tags$br(),
            tags$br()
          )
        )
      )
    )
  )

```

```

"The buttons that activate the application functions are
located on the left-side panel and the results will be
displayed on the main screen:",
tags$li("'File import' will import the original '.mat'
file. The file should be located on your
computer"),
tags$li("'Convert to csv' will convert the imported file to
'.csv' and 'save. csv' will save the converted file
on your computer"),
tags$li("'Load csv file' uploads the converted file to the
application" ),
tags$li("Select a subsegment of the uploaded file to
experiment with using the slider"),
tags$li("Load selected subsegment"),
tags$li("Plot 16 channel EEG for the selected subsegment"),
tags$li("Select and plot a channel of interest"),
tags$li("'Calculate and plot BDM' will estimate the
Kolmogorov-Chaitin complexity (KCC) of the
subsegment and plot it as univariate time
series"),
tags$li("Pressing 'WordCloud' button will generate and plot
the most common words (which encode KCC by a
number2letter function, e.g., 1 -> 'a', 2 -> 'b',
3 -> 'c', 4 -> 'd', etc. after rounding KCC values
to units) in the selected subsegment"),
tags$li("Pressing 'WordFrequencies' button will generate
and display a bar plot with frequencies of 10 most
common words"),
tags$li("'WordVectors' will display first 25 vectors (out
of 50 vectors of the word2Vec model built on
these EEG clips) for 10 the most frequent words
in the subsegment"),
tags$li("Selecting a sample from the drop-down menu and
pressing 'Classify segment' button will classify
the selected sample using a pretrained Word2Vec/
GBM model as preictal or interictal"),

tags$br(),
tags$br(),

"The main sources the system is based on include but are
not limited to:",

tags$li("Zenil, H.: A Review of Methods for Estimating
Algorithmic Complexity: Options, Challenges, and
New Directions. Entropy vol. 22(6) 612 (2020).
doi.org/10.3390/e22060612."),
tags$li("Zenil, H.: Towards Demystifying Shannon Entropy,
Lossless Compression, and Approaches to
Statistical Machine Learning. Proceedings of the
International Society for Information Studies 2019
summit, University of California, Berkeley, 47, 24
(2020). doi:10.3390/proceedings2020047024"),
tags$li("Mikolov, T. et al.: Efficient Estimation of Word
Representations in Vector Space. arXiv:301.3781v3,
7 Sep, 2013."),
tags$li("Mikolov, T., Sutskever, I., Chen, K., Corrado,
G.S., Dean, J.: Distributed representations of
words and phrases and their compositionality.
Proc. Adv. Neural Inf. Process. Syst. Nevada, NV
USA, pp. 3111-3119 (2013)"),

```



```

library(NCmisc)
library(plyr)

h2o.init(nthreads=-1, max_mem_size="8G")

shinyServer(function(input, output) {

  observeEvent(input$reset, {

    inFile1 <- NULL
    inFile2 <- NULL
    Fin_csv(NULL)
    Data_FF(NULL)
    Data_BDM(NULL)
    Data_d(NULL)
    output$EEG <- NULL
    output$Chan_EEG <- NULL
    output$BDM_TS <- NULL
    output$WORD_CLOUD <- NULL
    output$WORD_FR <- NULL
    output$Result0 <- NULL
    renderUI({})

  })

  data_mat <- reactive({

    inFile1 <- input$file1

    if (is.null(inFile1))
      return(NULL)

    mat_tbl <- readMat(inFile1$datapath)

    return(mat_tbl)

  })

  Fin_csv<- reactiveVal(NULL)

  observeEvent(input$action1,
    {
      data_converted<- convert_to_csv(data_mat())
      Fin_csv(data_converted)
    })

  output$downloadData <- downloadHandler(
    filename = function() {
      paste("Dog_2-data-", Sys.Date(), ".csv", sep="")
    },
    content = function(file) {
      write.table(Fin_csv(), file, sep = ",", row.names = F)
    }
  )

  data_csv <- reactive({

    inFile2 <- input$file2

```

```

    if (is.null(inFile2))
      return(NULL)

    csv_tbl <- read.csv(inFile2$datapath)

    return(csv_tbl)
  })

Data_FF <- reactiveVal(NULL)

observeEvent(input$action_s,
  {
    data_f<- as.data.frame(data_csv())
    data_ff<- as.data.frame(data_f[1:239760,])
    if(input$slider1 >= 1 & input$slider1 <= 35){
      lgth <- 6000
      e_row <- input$slider1 * lgth
      s_row <- e_row - 5999
      Data_F<- data_ff[s_row:e_row,]

    }else{

      lgth <- 5952
      e_row <- input$slider1 * lgth + 1680
      s_row <- e_row - 5951
      Data_F<- data_ff[s_row:e_row,]}

    Data_FF(Data_F)

  })

data_16<- reactive({

  req(Data_FF())
  data_f<- Data_FF()

  title<- paste("EEG trace for subsegment ", input$slider1, sep = " ")
  plot(1:17, xlim = c(0,6040), axes=F,ylab = "Channels", xlab = "Time"
    main = title)
  Axis(side=1, labels=T)
  grid(150, 100, lwd = 0.3, adjustcolor("green",alpha.f=0.2))
  grid(10,17, lwd = 1, "green")

  subplot(plot(data_f[["Ch1"]], type = "l", lwd = 0.3, bty="n", axes=
    FALSE, xlab = '', ylab = '1'), c(1,6026), c(16,17))

  subplot(plot(data_f[["Ch2"]], type = "l", lwd = 0.3, bty="n", axes=
    FALSE,xlab = '', ylab = '2'), c(1,6026), c(15,16))

  subplot(plot(data_f[["Ch3"]], type = "l", lwd = 0.3, bty="n", axes=
    FALSE,xlab = '', ylab = '3'), c(1,6026), c(14,15))

  subplot(plot(data_f[["Ch4"]], type = "l", lwd = 0.3, bty="n", axes=
    FALSE,xlab = '', ylab = '4'), c(1,6026), c(13,14))

  subplot(plot(data_f[["Ch5"]], type = "l", lwd = 0.3, bty="n", axes=
    FALSE,xlab = '', ylab = '5'), c(1,6026), c(12,13))

```

```

subplot(plot(data_f[["Ch6"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '6'), c(1,6026), c(11,12))

subplot(plot(data_f[["Ch7"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '7'), c(1,6026), c(10,11))

subplot(plot(data_f[["Ch8"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '8'), c(1,6026), c(9,10))

subplot(plot(data_f[["Ch9"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '9'), c(1,6026), c(8,9))

subplot(plot(data_f[["Ch10"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '10'), c(1,6026), c(7,8))

subplot(plot(data_f[["Ch11"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '11'), c(1,6026), c(6,7))

subplot(plot(data_f[["Ch12"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '12'), c(1,6026), c(5,6))

subplot(plot(data_f[["Ch13"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '13'), c(1,6026), c(4,5))

subplot(plot(data_f[["Ch14"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '14'), c(1,6026), c(3,4))

subplot(plot(data_f[["Ch15"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '15'), c(1,6026), c(2,3))

subplot(plot(data_f[["Ch16"]], type = "l", lwd = 0.3, bty="n", axes=
FALSE,xlab = '', ylab = '16'), c(1,6026), c(1,2))

})

observeEvent(input$getPlot1,
{
  output$EEG <- renderPlot({
    data_16()
  })
})

data2<- reactive({
  req(Data_FF())
  data_f<- Data_FF()

  time<- c(1:nrow(data_f))

  Channel<- as.character(paste("Ch", input$Channel, sep = ''))

  eegtime(time, data_f[[Channel]], flipvoltage = F, vltty = 1, vlwd = 0.5,
    vcol = "blue", voltageSE = NULL, slty = NA, slwd = 1,
    scol = "cyan", salpha = 0.65, conflevel = 0.95,
    plotzero = TRUE, zltty = 1, zlwd = 0.5, zcol = "black",
    xlim = NULL, ylim = NULL, xlab = NULL, ylab = NULL,
    nxtick = 6, nytick = 6, xticks = NULL, yticks = NULL,

```

```

        add = FALSE, main = "EEG for selected channel")
    })

observeEvent(input$getPlot2,
  {
    output$Chan_EEG <- renderPlot({
      data2()
    })
  })

Data_BDM<- reactiveVal(NULL)

data3<- reactive({
  req(Data_FF())
  data_f<- Data_FF()

  NumberOfCluster <- 2

  cl <- makeCluster(NumberOfCluster)
  registerDoSNOW(cl)

  fourByFourCTM <- read.csv("www/K-4x4.csv",
    stringsAsFactors=FALSE,
    colClasses =
      c("character", "numeric"),
    header = FALSE)

  colnames(fourByFourCTM) <- c("square", "CTM")
  rownames(fourByFourCTM) <- fourByFourCTM$square
  fourByFourCTM$square <- NULL

  require(purrr)
  ind <- function(matDim, blockSize, offset) {
    Map(`:` , seq(1, matDim-blockSize+1, by = offset),
      seq(blockSize, matDim, by = offset))
  }

  myPartition <- function(mat, blockSize, offset) {
    lapply(cross2(ind(nrow(mat),blockSize,offset),
      ind(ncol(mat),blockSize,offset)),
      function(i) mat[i[[1]], i[[2]])]
  }

  stringify <- function(smallBlock){
    paste0(c(t(smallBlock)), collapse = "")
  }

  bdm2D <- function(mat, blockSize, offset){
    parts <- myPartition(mat, blockSize, offset)

    flatSquares <- unlist(lapply(parts, stringify))

    squaresTally <- as.data.frame(table(flatSquares))
  }

```

```

rownames(squaresTally) <- squaresTally$flatSquares

squaresTally$flatSquares <- NULL

bdm <- (sum(fourByFourCTM[rownames(squaresTally),])
        + sum(log2(squaresTally$Freq)))
return(bdm)
}

a<- nrow(data_f)
b<- ncol(data_f)
c<- floor(a/b)

indx <- gl(c, b)

spt1 <- split(data_f, indx)

bin_mat_1<- foreach(i=1:length(spt1), .packages = "biclust") %dopar%
  binarize(spt1[[i]]) #, method = "BASCA")
bin_mat_f_1<- foreach(i=1:length(bin_mat_1)) %dopar%
  as.data.frame(bin_mat_1[[i]],nrow=16,ncol=16)
fin_1<- foreach(i=1:length(bin_mat_f_1), .combine = "c", .packages
  ="purrr") %dopar% bdm2D(bin_mat_f_1[[i]],4,4)
BDM_value<- ts(fin_1, start = 0, frequency = 1)

Data_BDM(Data_BDM_value)

plot.ts(BDM_value, col = "red", xlab="Time (one time-step is equivalent
  to 37.5 ms)", ylim = c(0,500), main = "BDM value over time")

stopCluster(cl)

})

observeEvent(input$action3,
{
  output$BDM_TS <- renderPlot({
    data3()
  })
})

Data_d<- reactiveVal(NULL)

data_word_cloud<- reactive({
  req(Data_BDM())

  data_m<- as.data.frame(Data_BDM())

  data_r<- round(data_m, digits = 0)
  data_l<-as.numeric(unlist(data_r))
  res <- lapply(data_l, number2Letter)
  res_b<- paste( unlist(res), collapse=',')

```

```

text_ff<- str_split_fixed(res_b, ",", 350)

docs <- Corpus(VectorSource(text_ff))
dtm <- TermDocumentMatrix(docs)
m <- as.matrix(dtm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)

Data_d(d)

set.seed(1234)
wordcloud(words = d$word, freq = d$freq, min.freq = 1,
          max.words= 20, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"),
          main ="Word Cloud for selected subsegment")
})

observeEvent(input$action4,
{

  output$WORD_CLOUD <- renderPlot({

    data_word_cloud()

  })
})

data_word_frequencies<- reactive({

  req(Data_d())

  data_wf<- as.data.frame(Data_d())

  barplot(data_wf[1:10,]$freq, las = 2, names.arg = data_wf[1:10,]$word,
          col ="lightblue", main ="Most frequent words in selected
          subsegment", ylab = "Word frequencies")

})

observeEvent(input$action5,
{

  output$WORD_FR <- renderPlot({

    data_word_frequencies()

  })
})

data_txt<- reactive({

  req(Data_d())

  dtm_words<- as.data.frame(Data_d()[1:10,])
  w_emb<- read.csv("www/Word_embeddings.csv")

  result<- match_df(w_emb, dtm_words, on="word")

  f_result<- result[, 1:26]

```

```

    print(f_result)
  })
observeEvent(input$action6,
  {
    output$Result0 <- renderPrint({
      data_txt()
    })
  })
sample_rnd<- reactive({
  sample_data<- read.csv("www/dl_test_as_vec.csv")
  s0<- sample_data[1:45,-1]
  s1<- sample_data[46:92,-1]
  if (input$select == 1 | input$select == 3 | input$select == 4) {
    samp1 <- sample(nrow(s0),1)
    selected <- s0[samp1,]
  }else{
    samp2 <- sample(nrow(s1),1)
    selected <- s1[samp2,]
  }
  return(selected)
})
data6<- reactive({
  sample_data<- read.csv("www/dl_test_as_vec.csv")
  s0<- sample_data[1:45,-1]
  s1<- sample_data[46:92,-1]
  new_sample<- sample_rnd()
  new_sample_c<- as.h2o(new_sample)
  path<- "www/GBM_model_R_1692355267008_105"
  Model_DL_loaded <- h2o.loadModel(path)
  pred <- as.data.frame(h2o.predict(object=Model_DL_loaded,
    newdata=new_sample_c))
  result<- as.data.frame(pred)
  result1<- as.matrix(result[1,1])
  if (result1 == 1){

```

```

      print("Class: preictal")
    }else{
      print("Class: interictal")
    }
  })

  observeEvent(input$action7,
    {
      output$Result <- renderText({
        data6()
      })
    })
})

```

Appendix 5. Certificates for inventions that formalize the study results



**REPUBLICA MOLDOVA
MINISTERUL SĂNĂTĂȚII**

**INSTITUȚIA MEDICO-SANITARĂ PUBLICĂ
INSTITUTUL DE MEDICINĂ URGENTĂ
(IMSP IMU)**

MD-2004, mun. Chișinău, str. T. Ciorba, 1



CERTIFICAT DE INOVATOR

Nr. 9

data 10 iunie 2026

de înregistrarea inovației în Registrul obiectelor proprietății intelectuale ale IMSP IMU în conformitate cu
art. 16 al Legii nr. 138-XV din 10.05.2001

**IMPLEMENTAREA UNUI MODEL CU ȘASE
COMPARTIMENTE AL SISTEMULUI CARDIOVASCULAR
ÎN MEDIUL NETLOGO PENTRU SIMULAREA
HEMODINAMICII ÎN CONDIȚII FIZIOLOGICE ȘI ÎN
INFARCTUL MIOCARDIC**

(denumirea)

IAPĂSCURTĂ V., BELÎI A.

(coautori)



Director IMSP IMU

MANEA Diana

mun. Chișinău



REPUBLICA MOLDOVA

MINISTERUL SĂNĂTĂȚII

INSTITUȚIA MEDICO-SANITARĂ PUBLICĂ

INSTITUTUL DE MEDICINĂ URGENTĂ

MD-2004, mun. Chișinău, str. T. Ciorba, 1

tel.: 022-23-78-84, fax: 022-23-53-09,

e-mail: anticamera@urgenta.md

www.urgenta.md

APROB

Director IMSP IMU

Diana MANEA

ACT DE IMPLEMENTARE nr. 9

1. Denumirea propunerii de implementare: **IMPLEMENTAREA UNUI MODEL CU ȘASE COMPARTIMENTE AL SISTEMULUI CARDIOVASCULAR ÎN MEDIUL NETLOGO PENTRU SIMULAREA HEMODINAMICII ÎN CONDIȚII FIZIOLOGICE ȘI ÎN INFARCTUL MIOCARDIC.**
2. **Autori:** IAPĂSCURTĂ V., conf. univ., dr. șt. med., BELÎI A., prof. univ., dr. hab. șt. med.
3. Numărul inovației: *Nr. 9 din 10 iunie 2026.*
4. **Unde și când a fost implementată:** Catedra de anesteziologie și reanimatologie nr. 1 „Valeriu Ghereg”, IP USMF „Nicolae Testemițanu”, și Departamentul Clinic Anestezie și Terapie Intensivă al IMSP IMU, anii 2022–2026.
5. **Rezultatul implementării:** Ca rezultat al elaborării și testării modelului cu șase compartimente, este pus la dispoziție un instrument computațional accesibil, care permite simularea hemodinamicii sistemului cardiovascular în diverse scenarii fiziologice și patologice. Modelul are potențial dublu de utilizare: ca prototip pentru asistarea deciziei clinice și ca resursă didactică interactivă în educația medicală.
6. **Eficacitatea implementării:** Modelul propus deschide noi perspective de cercetare în domeniul hemodinamicii și al ingineriei biomedicale, oferind o platformă flexibilă, ușor de extins și de validat ulterior cu date clinice. În ansamblu, se poate obține un efect benefic asupra calității cercetării și a pregătirii cadrelor medicale.

Prezenta inovație este implementată conform descrierii în cerere

Persoana responsabilă de implementare,
Viceirector de cercetare și transfer tehnologic
Dr.hab.șt.med., prof.univ.

Mihai CIOCANU



REPUBLICA MOLDOVA
MINISTERUL SĂNĂTĂȚII

INSTITUȚIA MEDICO-SANITARĂ PUBLICĂ
INSTITUTUL DE MEDICINĂ URGENTĂ
(IMSP IMU)

MD-2004, mun. Chișinău, str. T. Ciorba, 1



CERTIFICAT DE INOVATOR

Nr. _7_

data 09 iunie 2026

de înregistrarea inovației în Registrul obiectelor proprietății intelectuale ale IMSP IMU în conformitate cu
art. 16 al Legii nr. 138-XV din 10.05.2001

**IMPLEMENTAREA UNUI MODEL BAZAT PE AGENȚI ÎN
MEDIUL NETLOGO PENTRU SIMULAREA ȘI
VIZUALIZAREA FORMĂRII EDEMULUI PULMONAR
CARDIOGEN SUB INFLUENȚA PRESIUNILOR
HIDROSTATICĂ ȘI ONCOTICĂ**

(denumirea)

IAPĂSCURTĂ V.,BELÎI A.,

(coautori)



Director IMSP IMU

MANEA Diana

mun. Chișinău



REPUBLICA MOLDOVA

MINISTERUL SĂNĂTĂȚII

INSTITUȚIA MEDICO-SANITARĂ PUBLICĂ

INSTITUTUL DE MEDICINĂ URGENTĂ

MD-2004, mun. Chișinău, str. T. Ciorba, 1

tel.: 022-23-78-84, fax: 022-23-53-09,

e-mail: anticamera@urgenta.md

www.urgenta.md

APROB

Director IMSP IMU

Diana MANEA

ACT DE IMPLEMENTARE nr. 7

1. Denumirea propunerii de implementare: **IMPLEMENTAREA UNUI MODEL BAZAT PE AGENȚI ÎN MEDIUL NETLOGO PENTRU SIMULAREA ȘI VIZUALIZAREA FORMĂRII EDEMULUI PULMONAR CARDIOGEN SUB INFLUENȚA PRESIUNILOR HIDROSTATICĂ ȘI ONCOTICĂ.**
2. **Autori:** IAPĂSCURTĂ V., conf. univ., dr. șt. med., BELÎI A., prof. univ., dr. hab. șt. med.
3. **Numărul inovației:** Nr. 7 din 09 iunie 2026.
4. **Unde și când a fost implementată:** Catedra de anesteziologie și reanimatologie nr. 1 „Valeriu Ghereg”, IP USMF „Nicolae Testemițanu”, și Departamentul Clinic Anestezie și Terapie Intensivă al IMSP IMU, anii 2022–2026.
5. **Rezultatul implementării:** Ca rezultat al elaborării și testării modelului bazat pe agenți, este pus la dispoziție un instrument computațional accesibil, care permite simularea și vizualizarea formării edemului pulmonar cardiogen în diverse condiții. Modelul are potențial dublu de utilizare: ca resursă didactică interactivă în educația medicală și ca prototip pentru cercetarea fiziopatologiei edemului pulmonar.
6. **Eficacitatea implementării:** Modelul propus deschide noi perspective de cercetare în domeniul ingineriei biomedicale și al fiziopatologiei respiratorii, oferind o platformă flexibilă, ușor de extins (de exemplu prin adăugarea schimbului de gaze sau a permeabilității variabile a membranei) și de validat ulterior cu date clinice. În ansamblu, se poate obține un efect benefic asupra calității cercetării și a pregătirii cadrelor medicale.

Prezenta inovație este implementată conform descrierii în cerere.

Persoana responsabilă de implementare,
Viceirector de cercetare și transfer tehnologic
Dr.hab.șt.med., prof.univ.

Mihai CIOCANU



REPUBLICA MOLDOVA
MINISTERUL SĂNĂTĂȚII

INSTITUȚIA MEDICO-SANITARĂ PUBLICĂ
INSTITUTUL DE MEDICINĂ URGENTĂ
(IMSP IMU)

MD-2004, mun. Chișinău, str. T. Ciorba, 1



CERTIFICAT DE INOVATOR

Nr. 4

data 18 Mai 2026

de înregistrarea inovației în Registrul obiectelor proprietății intelectuale ale IMSP IMU în conformitate cu
art. 16 al Legii nr. 138-XV din 10.05.2001

**IMPLEMENTAREA UNUI MODEL HIBRID (BAZAT PE
AGENȚI ȘI DINAMICA SISTEMELOR) ÎN MEDIUL
NETLOGO PENTRU SIMULAREA INTERACȚIUNILOR
MATERNO-PLACENTO-FETALE ȘI A LIVRĂRII
OXIGENULUI LA FĂȚ**

(denumirea)

IAPĂSCURTĂ V.,BELÎI A.,

.(coautori)



Director IMSP IMU

MANEA Diana

mun. Chișinău



REPUBLICA MOLDOVA

MINISTERUL SĂNĂTĂȚII

INSTITUȚIA MEDICO-SANITARĂ PUBLICĂ

INSTITUTUL DE MEDICINĂ URGENTĂ

MD-2004, mun. Chișinău, str. T. Ciorba, 1

tel.: 022-23-78-84, fax: 022-23-53-09,

e-mail: anticamera@urgenta.md

www.urgenta.md

APROB

Director IMSP IMU

Diana MANEA

ACT DE IMPLEMENTARE Nr.4

1. Denumirea propunerii de implementare: **IMPLEMENTAREA UNUI MODEL HIBRID (BAZAT PE AGENȚI ȘI DINAMICA SISTEMELOR) ÎN MEDIUL NETLOGO PENTRU SIMULAREA INTERACȚIUNILOR MATERNO-PLACENTO-FETALE ȘI A LIVRĂRII OXIGENULUI LA FĂȚ**
2. Autori: IAPĂSCURTĂ V., conf. univ., dr. șt. med., BELÎI A., prof. univ., dr. hab. șt. med.
3. Numărul inovației: *Nr.4 din 18 Mai 2026.*
4. Unde și când a fost implementată: Catedra Anestezologie și reanimatologie „Valeriu Ghereg” IP USMF „Nicolae Testemițanu”, Departamentul Clinic Anestezie și Terapie Intensivă a IMSP IMU, anilor 2022-2026
5. Eficacitatea implementării: Ca rezultat al elaborării și testării modelului hibrid, este pus la dispoziție un instrument computațional accesibil public, care permite simularea interacțiunilor materno-placento-fetale și a livrării oxigenului la făt în diverse scenarii fiziologice și patologice. Modelul are potențial dublu de utilizare: ca prototip pentru asistarea deciziei clinice și ca resursă didactică interactivă în educația medicală.
6. Eficacitatea implementării: Modelul propus deschide noi perspective de cercetare în domeniul interacțiunilor materno-fetale și al ingineriei biomedicale, oferind o platformă flexibilă, ușor de extins și de validat ulterior cu date clinice. Integral, se poate obține un efect benefic asupra calității cercetării și a pregătirii cadrelor medicale.

Prezenta inovație este implementată conform descrierii în cerere

Persoana responsabilă de implementare,
Viceirector de cercetare și transfer tehnologic
Dr.hab.șt.med., prof.univ.

Mihai CIOCANU



REPUBLICA MOLDOVA
MINISTERUL SĂNĂTĂȚII

INSTITUȚIA MEDICO-SANITARĂ PUBLICĂ
INSTITUTUL DE MEDICINĂ URGENTĂ
(IMSP IMU)

MD-2004, mun. Chișinău, str. T. Ciorba, 1



CERTIFICAT DE INOVATOR

Nr. 20

data 14.12.2022

de înregistrarea inovației în Registrul obiectelor proprietății intelectuale ale IMSP IMU în
conformitate cu art. 16 al Legii nr. 138-XV din 10.05.2001

IMPLEMENTAREA APLICAȚIEI SOFTWARE ISAAC-EEG ÎN
BAZA COMPLEXITĂȚII ALGORITMICE PENTRU ANALIZA
ELECTROENCEFALOGRAAMEI

(denumirea)

IAPĂSCURTĂ Victor, FIODOROV Ion, LISNIC Vitalie

(coautori)



Director

dr. hab. șt. med., prof. univ. M. CIOCANU

mun. Chișinău



REPUBLICA MOLDOVA
MINISTERUL SĂNĂTĂȚII, MUNCII ȘI PROTECȚIEI SOCIALE
INSTITUȚIA MEDICO-SANITARĂ PUBLICĂ
INSTITUTUL DE MEDICINĂ URGENTĂ
MD-2004, mun. Chișinău, str. T. Ciorba, 1
tel.: 022-23-78-84, fax: 022-23-53-09,
e-mail: anticamera@urgenta.md
www.urgenta.md



APROB

Director IMSP IMU

Dr. hab șt. med., prof univ.,

M. CIOCANU

ACT DE IMPLEMENTARE

1. **Denumirea ofertei pentru implementare:** "IMPLEMENTAREA APLICAȚIEI SOFTWARE ISAAC-EEG ÎN BAZA COMPLEXITĂȚII ALGORITMICE PENTRU ANALIZA ELECTROENCEFALOGramei „
2. **Autori:** IAPĂSCURTĂ Victor, asistent universitar, doctorand; FIODOROV Ion, conf. univ., dr., LISNIC Vitalie, dr.hab. șt.med., prof.univ.
3. **Unde și când a fost implementată:** Anul implementării 2019-2022, Catedra de anesteziologie și reanimatologie "V. Ghereg", USMF "Nicolae Testemițanu" în cadrul IMSP IMU; Departamentul Anesteziologie și Trapie Intensivă IMSP IMU/ Secția Neuroreanimare
4. **Rezultatul implementării:** Ca rezultat al elaborării sistemului de analiză a EEG cu ajutorul complexității algoritmice și învățării automate se implementează o metodă nou de analiză a EEG cu potențial clinic în plan de predicție a acceselor epileptice eminente și cu efecte benefice asupra managementului epilepsiei.
5. **Eficacitatea implementării:** Analiza EEG cu ajutorul aplicației deschide noi perspective în cercetarea epilepsiei, precum și poate aduce beneficii directe pacienților care suferă de această boală, în primul rând prin alertarea lor timpurie despre un iminent acces epileptic. Integral se poate obține un efect de fortificare a tratamentului.

Prezenta inovație este implementată conform descrierii în cerere

Persoana responsabilă de implementare,

Șef Catedra de anesteziologie și reanimatologie nr. 1 "V. Ghereg",
IP USMF „Nicolae Testemițanu”
dr. hab. șt. med., prof univ. *Serghei Șandru*

Declarația privind asumarea răspunderii

(Disclaimer of liability)

Subsemnatul, declar că în teză a fost utilizată inteligența artificială (în calitate de obiect de studiu și instrument/metodologie), fiind un element important al cercetării, precum și epizodic – pentru perfecționarea textului (corectitudine, fluidă, etc.).

Declar pe răspundere personală că materialele prezentate în teza de doctorat habilitat sunt rezultatul propriilor cercetări și realizări științifice. Conștientizez că, în caz contrar, urmează să suport consecințele în conformitate cu legislația în vigoare.

Iapăscută Victor

Data:

(Translation: I hereby declare that artificial intelligence was utilized in the thesis (as both a subject of study and a tool/methodology), serving as a significant element of the research, as well as episodically to refine the text (regarding correctness, fluency, etc.).

I declare on my own responsibility that the materials presented in the habilitation thesis are the result of my own research and scientific achievements. I acknowledge that, should this prove otherwise, I shall bear the consequences in accordance with applicable legislation.

Name, Date)

Personal information

First name Victor
Last name Iapăscuță
Date of birth 27/01/1966
Nationality Moldovan
Gender Male
Phone number (+373) 69 10 57 13 (Mobile)
Address 2 Tiras street, Chișinău, 2004, Republic of Moldova
e-mail victor.iapascurta@doctorat.utm.md; victor.iapascurta@usmf.md
LinkedIn <https://www.linkedin.com/in/victor-iapascurta-16230b6/>
Position university lecturer



Work experience

September 1994 - current

University lecturer, Department of Anesthesia and Intensive Care, N. Testemițanu State University of Medicine and Pharmacy (165 Stefan cel Mare Blvd., Chisinau, Republic of Moldova, <https://anesteziologie1.usmf.md/ro>)

- “Anesthesia and Intensive Care” course for 5th-year students
- „Perioperative Medicine” course for 6th-year students (in English)

November 2018–current

Head of the research subgroup “AID & (patho)physiological and clinical end models” with the private research group “Algorithmic Information Dynamics,” Algorithmic Dynamics Lab, Karolinska Institute, Sweden/Oxford Immune Algorithmics, UK/Algorithmic Nature Group, Paris, France

April 2018–August 2018

Participant and winner at Complexity Explorer Spring Challenge 2018, Santa Fe Institute, New Mexico, USA

- Elaboration and implementation of decision-making models using agent-based modeling approach resumed in “Decision-making under uncertainty: a simple strategy vs bounded rationality and randomness” paper
- Participation in the Coding Tournament
<https://www.complexityexplorer.org/challenges/pages/solution-gallery>

December 2006 – November 2010

Director at International Partnership for Development

- Operational management of research and social entrepreneurship projects

September 1994 – December 2009

Anesthesiologist at the City Emergency Hospital, 1 Toma Ciorba street, Chisinau, Moldova

September 1994 – December 2006

Program coordinator, Program director, President at the Medical Foundation of Moldova, 1 Toma Ciorba str., Chișinău, Moldova

- Operational management of technical assistance for education and medicine in partnership with government and private organizations from USA and Europe

Education and training <i>Qualification/Diploma</i> <i>Institution</i> <i>Qualification/Diploma</i> <i>Institution</i> Other training	September 1987 – June 1993 General practitioner, diploma E nr. 003648 from 30.06.1993 N. Testemițanu State University of Medicine and Pharmacy, 165 Stefan cel Mare bvd, Chisinau, Republic of Moldova August 1993 – June 1994 Anesthesia and Intensive Care Physician, certificate nr. 223 from 23.06.1994 N. Testemițanu State University of Medicine and Pharmacy, 165 Stefan cel Mare Blvd., Chisinau, Republic of Moldova/City Emergency Hospital, 1 Toma Ciorba street, Chisinau, Moldova
01.09-30.11.2022 01.06-05.10.2018 04.05-05.09.2017 04.05-05.09.2017 01.04-27.08.2015	✓ Deep Representation Learning • A course in the domain of Representation Learning by Nistor Grozavu, CY Cergy Paris University as part of doctoral thesis research program ✓ Algorithmic Information Dynamics: A Computational Approach to Causality and Living Systems From Networks to Cells • A course in the domain of algorithmic information dynamics, Complexity Explorer, Santa Fe Institute, New Mexico, USA ✓ Introduction to Agent-Based Modeling (2017) • A course in the domain of agent-based modeling and Net Logo programming, Complexity Explorer, Santa Fe Institute, New Mexico, USA ✓ Introduction to Dynamical Systems and Chaos (2017) • A course in the domain of Dynamic Systems, Complexity Explorer, Santa Fe Institute, New Mexico, USA Sleep Neurobiology, Medicine and Society • A course in sleep medicine from University of Michigan, USA
Conference participation (selective)	✓ AUTOMATA 2020, 10-12 august, Stockholm, Sweden • Report: „Block Decomposition Method and Traditional Machine Learning for Epileptic Seizure Prediction” ✓ 5th International Conference on Nanotechnologies and Biomedical Engineering, November 3-5, 2021, Chisinau, Moldova, • Report: A Less Traditional Approach to Biomedical Signal Processing for Sepsis Prediction, Springer IFMBE Proceedings, doi:10.1007/978-3-030-92328-0_29; • Report: Preclinical Stage of Building a Machine Learning System for Sepsis Prediction: A Comparative Study of Four Algorithms. Springer IFMBE Proceedings, doi:10.1007/978-3-030-92328-0_58. ✓ 6th International Conference on Nanotechnologies and Biomedical Engineering, September 20-23, 2023,, Chisinau, Moldova, • Report: A Less Common Algorithmic Complexity Approach to EEG Signal Processing for Machine Learning, Springer IFMBE Proceedings, 2024. https://doi.org/10.1007/978-3-031-42782-4_20; • Report: NLP Tools for Epileptic Seizure Prediction Using EEG Data: A Comparative Study of Three ML Models. Springer IFMBE Proceedings 92, https://doi.org/10.1007/978-3-031-42782-4_19.
Research specialty Research domain The topic of doctoral thesis	122.03. Modeling, Mathematical Methods, Software Products; 321.19. Anesthesia and Intensive Care. Machine learning/Artificial Intelligence in Healthcare Intelligent Methods and Algorithms for Predicting the Clinical State of the Patient

Projects 2019 2020 - current	Algorithmic Information Dynamics for the analysis of Physiological Models Algorithmic Information Dynamics for Clinical Purposes																				
Publications (selective)	- New approaches to missing biomedical data recovery for machine learning. Journal of Engineering Science 2023, 30(1), pp. 106-117. (coauthored with Fiodorov I.) https://doi.org/10.52326/jes.utm.2023.30(1).09 - Combining Algorithmic Information Dynamics Concepts and Machine Learning for Electroencephalography Analysis: What Can We Get?. In: Complex Systems Journal. 2022, 31(4), pp. 389-413. https://doi.org/10.25088/ComplexSystems.31.4.389 - Detection of Movement toward Randomness by Applying the Block Decomposition Method to a Simple Model of the Circulatory System. In: Complex Systems Journal. 2019, 28(1), pp. 59-77. https://doi.org/10.25088/ComplexSystems.28.1.59																				
Honors and awards	- Defended the doctoral thesis in medical science with “Summa Cum Laude” (2023) - Winner of the Coding Tournament (1st place), Complexity Explorer Spring Challenge 2018, Santa Fe Institute, New Mexico, USA (2018) 2nd place for project paper (“Decision-making under uncertainty: a simple strategy vs bounded rationality and randomness”) at Complexity Explorer Spring Challenge 2018, Santa Fe Institute, New Mexico, USA (2018) - Graduated N. Testemitanu State University of Medicine and Pharmacy with “Summa Cum Laude” (1993)																				
Language skills Mother tongue Other languages: Russian English Digital skills	Romanian <table><tr><th colspan="2">Understanding</th><th colspan="2">Speaking</th><th>Writing</th></tr><tr><td>Listening</td><td>Reading</td><td>Spoken Production</td><td>Spoken Interaction</td><td>Writing</td></tr><tr><td>C2</td><td>C2</td><td>C2</td><td>C2</td><td>C2</td></tr><tr><td>B2</td><td>B2</td><td>B2</td><td>B2</td><td>B2</td></tr></table> MS Office applications: Word, Excel, and PowerPoint; Internet, R/RStudio, Octave, Wolfram Mathematica, H2O, Net Logo, Python	Understanding		Speaking		Writing	Listening	Reading	Spoken Production	Spoken Interaction	Writing	C2	C2	C2	C2	C2	B2	B2	B2	B2	B2
Understanding		Speaking		Writing																	
Listening	Reading	Spoken Production	Spoken Interaction	Writing																	
C2	C2	C2	C2	C2																	
B2	B2	B2	B2	B2																	
Driving license	Driving license: B																				
Recommendations	- Fiodorov Ion, dr. associate professor, ion.fiodorov@ati.utm.md, phone (+373) 79454240. - Belii Adrian, dr. hab. in med. sci., professor, adrian.belii@usmf.md, phone (+373) 79579474.																				