People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
Mohamed El Bachir El Ibrahimi University of Borj Bou Arréridj
Faculty of Mathematics and Computer Science
Department of Computer Science



THESIS

In order to obtain the Doctorate degree in LMD (3rd cycle)
Branch: Computer Science
Option: Ingénierie de l'informatique décisionnelle

THEME

Narrative Approaches and Machine Learning for Health Knowledge Management: Themes Extraction and Texts Classification

By: **Hafida Tiaiba**

Publicly defended on: 23/09/2025

In front of the jury composed of:

Pr. MOUSSAOUI Abdelouahab	**University of Sétif 1**	**President**
Dr. SABRI Lyazid	**University of Bordj Bou Arreridj**	**Supervisor**
Pr. CHIBANI Abdelghani	**University of UPEC France**	**Co-supervisor**
Pr. LAALLAM Fatima Zohra	**University of Ouargla**	**Examiner**
Pr. REZEG Khaled	**University of Biskra**	**Examiner**
Pr. ATTIA Abdelouahab	**University of Bordj Bou Arreridj**	**Examiner**
Dr. CHARIKHI Mourad	**University of Bordj Bou Arreridj**	**Examiner**

2024/2025

Acknowledgments

First and foremost, I would like to express my deepest gratitude to God, whose infinite blessings, guidance, and strength have supported me throughout every step of this journey.

I wish to express my profound and heartfelt appreciation to Professor Lyazid Sabri. His exceptional mentorship, insightful advice, and continuous encouragement have been a guiding light throughout my research. His visionary thinking, dedication to scientific rigor, and generous support have deeply influenced the quality and direction of this thesis. It has been a true privilege to learn under his supervision. I would like to express my sincere gratitude to Professor Abdelghani Chibani for his valuable support and technical guidance at the early stages of this research. I also extend my sincere thanks to Professor Okba Kazar for his valuable guidance, constructive feedback, and steadfast encouragement. His critical insights and support have significantly contributed to the development and enrichment of this work.

I would like to express my sincere gratitude to all the esteemed members of the thesis jury for their valuable time. I extend my deepest appreciation to Prof. Abdelouahab Moussaoui, President of the jury from the university of Sétif 1. Special thanks go to my thesis director Lyazid Sabri from the university of Bordj Bou Arreridj, whose unwavering support and supervision have been instrumental throughout the development of this research. I am also thankful to the examiners, Prof. Fatima Zohra Laallam from the university of Ouargla and Prof. Khaled Rezeg from the university of Biskra. Furthermore, I express my appreciation to the examiners, Prof. Abdelouahab Attia and Dr. Mourad Charikhi from the university of Bordj Bou Arreridj. Finally, I warmly thank Prof. Abdelghani Chibani from the University of UPEC, France, for honoring us with his presence as an invited member of the jury.

I am grateful to all the faculty members and staff of the Department of Computer Science for fostering an environment of knowledge, innovation, and academic excellence. My heartfelt thanks also go to Dr. Ghania Chellakh Dental Surgeon and Dr. Latifa Benadelsalem general practitioner for their valuable assistance in understanding and validating some medical texts cases; and My sincere thanks to Dr. Fatma Noui, cardiologist. I would also like to sincerely thank my friends, Prof. Samia Aitouche and Dr. Malika Noui, for their encouragement

throughout this experience.

Finally, I extend my deepest gratitude to my beloved mother, whose unconditional love, prayers, and encouragement have been a constant source of strength throughout my journey. I am equally thankful to my dear father, whose wisdom, support, and unwavering belief in my potential have been instrumental in all my achievements. I would also like to express my heartfelt appreciation to my husband and to my two wonderful sons youness and abdelmounaim, who are my greatest source of inspiration and motivation.

This work is dedicated to all those who believe in the power of knowledge and the endless possibilities of scientific exploration.

Abstract

The healthcare sector generates a massive volume of heterogeneous information every day, a large portion of which takes a narrative form, such as medical records, hospitalisation reports, and patient testimonies. In the context of exponential growth in textual medical data, it has become essential to transform this information into machine-readable knowledge to support personalised diagnoses, therapeutic monitoring, and the practice of precision medicine. This thesis proposes an interdisciplinary approach that combines symbolic artificial intelligence, machine learning—including deep learning techniques—and medical ontologies to enhance the representation, organisation, and utilisation of healthcare knowledge. Ontologies structure and provide meaning to specific terms, abbreviations, and contextual dependencies present in medical texts, while advanced machine learning techniques facilitate their automatic classification into disease categories, treatment plans, or predefined diagnostic groups. The integration of these methods provides a robust framework for managing narrative knowledge in healthcare, optimising the retrieval of relevant information, supporting clinical decision-making, and enabling treatment planning tailored to individual patient needs. Thus, combining narrative formalisms, statistical methods, and ontological resources offers a powerful means to fully exploit the potential of textual medical data.

Keywords: Medical unstructured text, text mining, ontology, classification, embedding, machine learning, natural language processing.

Résumé

Le secteur de la santé génère chaque jour un volume massif d'informations hétérogènes, dont une grande partie prend forme narrative, comme les dossiers médicaux, les comptes rendus d'hospitalisation et les témoignages de patients. Face à la croissance exponentielle des données médicales textuelles, il est devenu essentiel de transformer ces informations en connaissances lisibles par machine afin de soutenir le diagnostic personnalisé, le suivi thérapeutique et la pratique de la médecine de précision. Cette thèse propose une approche interdisciplinaire combinant intelligence artificielle symbolique, apprentissage automatique (incluant les techniques d'apprentissage profond) et ontologies médicales pour améliorer la représentation, l'organisation et l'utilisation des connaissances en santé. Les ontologies structurent et donnent du sens aux termes, abréviations et dépendances contextuelles spécifiques présents dans les textes médicaux, tandis que les techniques avancées d'apprentissage automatique facilitent leur classification automatique en catégories de maladies, plans de traitement ou groupes diagnostiques prédéfinis. L'intégration de ces méthodes fournit un cadre robuste pour la gestion des connaissances narratives en santé, optimisant la recherche d'informations pertinentes, facilitant la prise de décision clinique et permettant une planification thérapeutique adaptée aux besoins de chaque patient. Ainsi, la combinaison de formalismes narratifs, de méthodes statistiques et de ressources ontologiques offre un moyen puissant d'exploiter pleinement le potentiel des données médicales textuelles.

Mots clés: Texte médical non structuré, fouille de texte, ontologie, encodage sémantique, apprentissage automatique, traitement automatique du langage naturel.

ملخّص

يُنتج قطاع الصحة يوميًّا حجمًا ضخمًا من المعلومات غير المتجانسة، يشكّل السرد جزءًا كبيرًا منها، مثل السجلات الطبية، وتقارير الاستشفاء، وشهادات المرضى. ومع النمو المتسارع للبيانات الطبية النصيّة، أصبح من الضروري تحويل هذه المعلومات إلى معارف قابلة للقراءة آليًّا، وذلك لدعم التشخيص الشخصي، ومتابعة العلاجات، وتطبيق ممارسات الطب الدقيق.

تقترح هذه الأطروحة مقاربةً متعددة التخصصات تجمع بين الذكاء الاصطناعي الرمزي، والتعلّم الآلي (بما في ذلك تقنيات التعلّم العميق)، والأونتولوجيات الطبية، من أجل تحسين تمثيل المعرفة الصحية وتنظيمها واستعمالها. وتوفّر الأونتولوجيات بنية ومعنى للمصطلحات والاختصارات والاعتماديات السياقية الخاصة بالنصوص الطبية، بينما تُسهّل تقنيات التعلّم الآلي المتقدّمة تصنيفها التلقائي ضمن فئات الأمراض، وخطط العلاج، أو المجموعات التشخيصية المحددة مسبقًا.

إن دمج هذه الأساليب يوفّر إطارًا متينًا لإدارة المعارف السردية في مجال الصحة، مما يُحسّن من البحث عن المعلومات ذات الصلة، ويدعم اتخاذ القرار السريري، ويتيح تخطيطًا علاجيًا متكيّفًا مع احتياجات كل مريض. وهكذا، فإن الجمع بين الصيغ السردية، والأساليب الإحصائية، والموارد الأنطولوجية يشكّل وسيلة قوية لاستغلال الإمكانات الكاملة للبيانات الطبية النصيّة.

الكلمات المفتاحية: النص الطبي غير المنظم، تنقيب النصوص، الأنطولوجيا، تضمين المعاني، التعلم الآلي، المعالجة الآلية للغة الطبيعية.

Contents

Acknowledgments **ii**

Abstract **iii**

Résumé **iv**

ملخّص **v**

List of Figures **xi**

List of Tables **xii**

Abbreviations List **xv**

General Introduction **1**

Motivation 1
Problematic 3
Research Issues and Objectives 4
Contributions 5
Organization of the Manuscript 6

1 Foundations of Machine Learning and Ontologies **8**

1.1 Introduction 9
1.2 Clinical Data: Nature and Structure 9
1.2.1 Clinical Data 9
1.2.2 Clinical Narrative 10
1.2.3 The Challenges 10
1.3 Core Concepts of Machine Learning 11
1.3.1 Introduction 11
1.3.2 Statistical Learning Models 12

1.3.3 Artificial Neural Networks and Their Deep Variants 17
1.4 Fundamentals of Ontologies 28
1.4.1 Introduction 28
1.4.2 OWL Ontologies 30
1.4.3 Uses in Healthcare 30
1.4.4 Resource Description Framework (RDF) 31
1.4.5 Relationship between RDF and OWL 31
1.4.6 Challenges and Limitations of RDF and OWL Ontologies 32
1.4.7 Jena API 33
1.4.8 Owlready2 Library 35
1.4.9 Usage and Applications 36
1.5 Conclusion 37

2 State of the Art of Themes Extraction **38**
2.1 Introduction 38
2.2 Information Extraction 39
2.3 Events, Temporal Expression, and TLink 40
2.4 Detection of Drugs Efficiency 43
2.5 Clinical Documents Classification 44
2.6 Text Vectorization 45
2.7 Natural Language Preprocessing 46
2.8 Ontologies in healthcare 47
2.9 Conclusion 48

3 New approachs for Events, TIMEX and Tlink extraction **50**
3.1 Introduction 51
3.2 New DL Model for Events and TIMEX extraction 51
3.2.1 Data Preprocessing 53
3.2.2 BERT Embeddings 56
3.2.3 Models Description 57
3.2.4 Evaluation Metrics 61
3.2.5 Results 62
3.2.6 Prediction Process 69
3.3 An advanced approach using Ontologies and DL for Events and TIMEX extraction 70
3.3.1 Ontologies Description 70
3.3.2 Key Biomedical and Clinical Entity Types 72

3.3.3 Ontologies using Jena API and Deep Learning . . . 73
3.3.4 Ontologies using Owlready2 and DL . . . 91
3.3.5 Jena API vs. Owlready2 . . . 103
3.4 New Approach for temporal Relations Extraction . . . 104
3.4.1 Dataset and Pre-Processing Process . . . 105
3.4.2 Models Description . . . 106
3.4.3 Results and discussion . . . 108
3.4.4 Temporal Link (TLink) Prediction Approach . . . 113
3.5 Conclusion . . . 118

4 The design of Clinical Documents Classification model 119
4.1 Introduction . . . 119
4.2 The Description of Ohsumed Dataset . . . 120
4.3 Proposed Models . . . 121
4.3.1 Numerical Text Representation Methods . . . 123
4.3.2 Artificial Neural Network (ANN) Architectures . . . 124
4.4 Results and Discussion . . . 124
4.5 Conclusion . . . 127

5 A novel framework for Detection of Drugs Efficiency 129
5.1 Introduction . . . 129
5.2 Used Dataset . . . 130
5.3 Data Preprocessing Strategies . . . 130
5.3.1 First Strategy: Minimal Text Transformation . . . 130
5.3.2 Second Strategy: Lemmatization Approach . . . 131
5.4 Numerical Text Representation . . . 132
5.5 Proposed Models . . . 132
5.5.1 Traditional Machine Learning Models . . . 132
5.5.2 Neural Network Models . . . 132
5.6 Results and Discussion . . . 135
5.6.1 Analysis of Model Performance without Lemmatization . . . 135
5.6.2 Analysis of Model Performance with Lemmatization . . . 136
5.7 Conclusion . . . 139

General Conclusion 140
Conclusion . . . 140
Future Works . . . 141

Valorization and Applications 143

LIST OF PUBLICATIONS **160**

Journals' papers 160

International Conferences 160

List of Figures

1.1 Basic architecture of an Artificial Neural Network. 18
1.2 Perceptron architecture. 19
1.3 The foundational mechanism of standard CNN 23
1.4 The foundational mechanism of transposed CNN 24
1.5 The basic mechanism of seperable CNN 25
1.6 ConvLSTM's structure internal 27

3.1 Train and Test datasets for i2b2-2012 coprpus 53
3.2 Test datasets for i2b2-2010 and ci2b2-2018 coprpora 54
3.3 Description of data preprocessing 55
3.4 The architecture of the basic model 59
3.5 A general description of entities extraction methodology 60
3.6 Models training progress with BERT_768 65
3.7 Models training progress with BERT_128 66
3.8 A comprehensive view of the DOID ontology 77
3.9 The results using ontologies with the Jena API 85
3.10 Entities extraction using medical ontologies and Jena API 86
3.11 Entities extraction using the CONSEPTR model 86
3.12 Entities extraction using ontologies and deep learning 87
3.13 Annotated XML file generated using the Jena API and CONSEPTR. 88
3.14 Inferences of 'pulmonary artery choriocarcinoma' and 'respiratory system' in the DOID 89
3.15 Inferences for the entities 'pulmonary artery choriocarcinoma' and 'respiratory system' in the DOID-Continuation- 90
3.16 The results using ontologies with the Owlready2 library 96
3.17 Entities extracted using medical ontologies 97
3.18 Entities extracted using DL (CONSEPTR model) 98
3.19 Entities extracted using ontologies and DL 98

3.20 Annotated XML file generated using the Owlready2 and CONSEPTR 99
3.21 The inferred subclasses resulting from the description logic query expression 11 100
3.22 Continuation of inferred subclasses, Figure 3.21 101
3.23 Inferred results of 'nutrient deficiency' and ('heart disease' and 'nutrient deficiency') 103
3.24 Fundamental temporal relationships defined by Allen's interval algebra . . . 105
3.25 Initial TLink distribution in train and test Data 107
3.26 TLink distribution after merging train and test datasets 108
3.27 Sample size by relation type for R_CONSEPTRL_8 model 111
3.28 F1-Score by relation type for R_CONSEPTRL_8 model 112
3.29 Performance comparison for relation extraction 114
3.30 XML file displaying detailed annotations of a patient's events, temporal expressions, and temporal relation 116

4.1 Distribution of training and testing datasets (Ohsumed) 121
4.2 Performance comparison with previous studies for diseases classification . . . 126

5.1 Train and test datasets for Drugs.com corpus after preprocessing 131
5.2 Comparison results of Embedding-100 model with previous studies results . 138

List of Tables

2.1 Summary of the key studies using i2b2-2012 dataset 42

3.1 Events and temporal expressions results for i2b2-2012 dataset using DL . . . 63
3.2 Results by type for i2b2-2012 dataset 64
3.3 Test results of the CONSEPTR model 68
3.4 The test results using the i2b2-2012 dataset (Jena API) 82
3.5 The test results using the i2b2-2010 dataset (Jena API) 82
3.6 The test results using the i2b2-2018 dataset (Jena API) 83
3.7 Comparison of previous studies and the proposed models 84
3.8 The performance results using i2b2-2012 (owlready2) 93
3.9 The performance results using i2b2-2010 (owlready2) 94
3.10 The performance results using i2b2-2018 (owlready2) 94
3.11 Comparative summary of DL models and ontology-integrated approaches . . 104
3.12 Global performance results for TLink models on the training and test datasets 109
3.13 Detailed results of relation extraction 110
3.14 Test results for the models R_CONSEPTRL_8 and R_CONSEPTRL_3 . . 110

4.1 The architecture of the models SNN and VNN. 125
4.2 Synthesis of the best results for diseases classification 126

5.1 Summary of Simple10 model architecture. 133
5.2 Summary of the Embedding-32- model architecture. 133
5.3 Summary of the Embedding-32-Maxpooling model architecture. 134
5.4 Summary of the Embedding-100 model strcture. 134
5.5 Summary of the CNN-Maxpooling model strcture. 134
5.6 Performance of models without applying lemmatization 136
5.7 Performance of models with applying lemmatization 137

List of Algorithms

1 Train and test the modular models and CONSEPTR model 60
2 Test's performance using Jena API and DL 79
3 Entity prediction via Jena API and DL 80
4 Pseudo algorithm prediction using Owlready2 92
5 Data preparation for TLink 106
6 Temporal relation extraction using R_CONSEPTRL_8 Model 115
7 Pseudo algorithm for medical texts classification 122

Abbreviations List

Abbreviation	Definition
Adam	Adaptive Moment Estimation
AI	Artificial Intelligence
ANN	Artificial Neural Network
Att-BiLSTM	Attention-based Bidirectional Long Short-Term Memory network
BERT	Bidirectional Encoder Representations from Transformers
BERT_128	bert_en_uncased_L-2_H-128_A-2/1
BERT_768	BERT_en_uncased_L-2_H-768_A-12/1
BOW	Bag of Words
CART	Classification and Regression Trees
CBOW	Continuous Bag of Words
ChEBI	Chemical Entities of Biological Interest
CNN	Convolutional Neural Networks
CRF	Conditional Random Field
DISDRIV	Disease Drivers ontology
DL	Deep Learning
DNN	Deep Neural Network
DOID	Human Disease Ontology
DRON	Drugs Ontology
DT	Decision Trees
DTM	Document-Term Matrix
EHR	Electronic Health Records
EMR	Electronic Medical Records
FCNN	Fully Connected Neural Network
FoodOn	food ontology
HIPAA	Health Insurance Portability and Accountability Act

IE	Information extraction
KNN	K-Nearest Neighbors
LSTM	Long Short-Term Memory
MeSH	Medical Subject Headings
ML	Machine Learning
MLP	Multilayer Perceptron
NER	Named Entity Recognition
NLM	National Library of Medicine
NLP	Natural Language Processing
LR	Logistic Regression
LSTM	Long Short-Term Memory
OBO	Open Biological and Biomedical Ontology
OLS	Ordinary Least Squares
OOB	out-of-bag
OWL	Web Ontology Language
PCA	Principal Component Analysis
RDFS	Resource Description Framework Schema
ReLU	Rectified Linear Unit
RF	Random Forests
RNNs	Recurrent Neural Networks
SGD	Stochastic Gradient Descent
SVC	Support Vector Classification
SVMs	Support Vector Machines
SYMP	Symptoms Ontology
TF	Term Frequency
TF-IDF	Term Frequency-Inverse Document Frequency
TIMEX	Temporal expressions
UBERON	Uber-anatomy ontology
URI	Uniform Resource Identifiers
W3C	World Wide Web Consortium

General Introduction

Contents

Motivation 1

Problematic 3

Research Issues and Objectives 4

Contributions 5

Organization of the Manuscript 6

This chapter lays the groundwork for the present research by outlining the underlying motivations and examining the core problematics addressed. It further defines the objectives that guide the study. Additionally, the chapter highlights the key contributions of the work and concludes with a structured overview of the manuscript's organization.

Motivation

The expanding use of Electronic Medical Records (EMRs) has transformed modern healthcare by providing a comprehensive digital repository of patient information. EMRs contain structured and unstructured data, including medical history, treatment plans, diagnostic reports, and clinical notes. They serve as a crucial resource for healthcare professionals, allowing them to track disease progression, assess treatment efficacy, and make well-informed clinical decisions. However, despite the vast amount of valuable information embedded within EMRs, leveraging free-text clinical narratives remains a significant challenge as a result of the linguistic complexity in clinical narratives and the limitations of existing text-mining techniques.

Medical text classification plays a crucial role in organizing and interpreting clinical narratives by automatically categorizing them into predefined disease categories, treatment plans, and diagnostic groups. This process facilitates clinical decision support, enhances predictive analytics, and improves patient care management. Furthermore, beyond structured medical

records, patient-generated content, such as drug reviews and treatment experiences, provides valuable real-world evidence regarding drug efficacy, side effects, and treatment outcomes. Analyzing these subjective accounts can help researchers and healthcare providers identify patterns of medication adherence, detect adverse drug reactions, and assess the effectiveness of pharmaceutical interventions beyond the controlled settings of clinical trials.

Despite the potential benefits of automated text classification and information extraction, several challenges persist. Clinical narratives often contain domain-specific terminologies, abbreviations, and contextual dependencies that complicate traditional text-mining approaches. Additionally, extracting events, temporal rexpressions, and temporal relations from EMRs is essential for constructing structured patient timelines and understanding the chronological progression of diseases and treatments. However, handling inconsistencies, contradictions, and missing data remains an unresolved issue that affects the reliability of extracted knowledge.

Recent advancements in Machine Learning (ML), and in particular Deep Learning (DL) have demonstrated remarkable potential in addressing these challenges. Among various DL architectures, Convolutional Neural Networks (CNNs) (originally developed for image processing) have been effectively adapted for NLP tasks such as medical relation classification, clinical text categorization, and structured data extraction. Previous research has successfully applied CNNs to classify medical relations [1], analyze nursing-care texts [2], and extract structured data from unstructured pathology reports [3]. However, while CNNs effectively capture local patterns in text, they have limitations in modeling long-range dependencies and understanding temporal relationships within medical narratives.

To improve event and temporal expression extraction, ontologies provide a structured approach to organizing medical knowledge. Unlike DL models that rely solely on statistical patterns, ontologies define explicit relationships between medical concepts, ensuring consistency in extracted information. This thesis incorporates ontologies, including the drugs ontology (DRON)[1], symptoms ontology (SYMP)[2], human disease ontology (DOID)[3], The Uber-anatomy ontology (UBERON)[4], food ontology (FoodOn)[5], and Disease Drivers ontology (DISDRIV)[6] to enhance the extraction of clinical events from text. This integration enables a more structured and comprehensive representation of patient histories.

[1] https://obofoundry.org/ontology/dron.html
[2] https://obofoundry.org/ontology/symp.html
[3] http://purl.obolibrary.org/obo/doid.owl
[4] https://www.ebi.ac.uk/ols4/ontologies/uberon
[5] https://www.ebi.ac.uk/ols4/ontologies/foodon
[6] https://www.ebi.ac.uk/ols4/ontologies/disdriv

Problematic

The increasing digitization of healthcare records, particularly through EMRs, has led to the accumulation of vast quantities of clinical narratives. These unstructured texts offer rich insights into patient histories, diagnoses, treatments, and outcomes. However, leveraging this data effectively remains a significant challenge due to its linguistic complexity, variability, and the prevalence of domain-specific terminologies and notations. A central difficulty lies in converting these narratives into machine-readable formats through text vectorization. This transformation is essential for applying ML models to tasks such as information extraction, classification, and predictive analytics. The accuracy of such models, however, is highly sensitive to both the choice of vectorization methods and the preprocessing strategies employed prior to vectorization.

Among the various preprocessing techniques, lemmatization may not always substantially enhance the performance of machine learning algorithms in pattern recognition tasks. For example, reducing terms such as ”diagnoses,” ”diagnosed,” and ”diagnosing” to their root form ”diagnose” could lead to a loss of meaningful distinctions, as each form conveys a specific and context-dependent meaning. Nevertheless, such preprocessing steps may inadvertently impact critical information, for example, by eliminating numerical data such as drug dosages or measurement values. Moreover, stemming can compromise the semantic integrity of the text, potentially leading to a degradation in the quality of analytical tasks. Despite the richness of EMRs, their full potential is underutilized due to several interconnected challenges:

1. **The complexity of event extraction:** Clinical narratives often contain unstructured, noisy, and heterogeneous data, which complicates the extraction of events, particularly medical events.
2. **Difficulties in temporal information extraction from clinical narratives:** Extracting temporal relations from clinical narratives is a crucial step in building coherent patient timelines and enabling chronological reasoning in medical decision-making. However, this task remains challenging due to the ambiguity of temporal expressions, variability in narrative styles, and the frequent implicitness of temporal relationships. These factors complicate the ability of computational models to accurately identify, interpret, and classify temporal links, thereby reducing the overall reliability of automated temporal reasoning systems in clinical contexts.
3. **Preprocessing sensitivity:** Preprocessing steps such as symbol elimination, lemmatization, and numerical data removal can significantly influence the performance of machine learning models. However, these operations also carry the risk of discarding

essential contextual or domain-specific information, which may negatively affect the model's effectiveness.

4. **Vectorization limitations:** The effectiveness of various text vectorization techniques in preserving medical semantics and syntactic structures varies considerably, which can directly impact the accuracy and interpretability of classification models.
5. **Consumer-Generated data integration:** Incorporating insights from drug review datasets such as Drugs.com introduces informal language and non-standard expressions, which pose additional challenges for preprocessing and classification.

Research Issues and Objectives

Clinical narratives contain rich but unstructured information on patient status, interventions, and temporal evolution. This thesis therefore concentrates on three inter-dependent tasks—event extraction, temporal expression recognition, and temporal relation detection—and investigates how deep-learning models can be strengthened through the explicit integration of domain ontologies. The research issues identified lead to the following objectives:

1. **Ontology-Augmented deep learning:** Quantify the contribution of domain ontologies to semantic comprehension when integrated with DL. additionally, exploit the capabilities of the Jena API and Owlready2 for efficient management, querying over domain-specific medical ontologies.
2. **Joint extraction of events, temporal expressions, and temporal relations:** Design models capable of (i) identifying clinical events (e.g., *Problem*, *Test*, *Treatment*), and *Clinical_Dept*, *Evidential*, and *Occurrence*. (ii) recognising temporal expressions (date, time, duration, frequency), and (iii) inferring temporal links (*ENDED_BY*, *BEGUN_BY*, *DURING*, *BEFORE_OVERLAP*, *SIMULTANEOUS*, *AFTER*, *OVERLAP*, and *BEFORE*) within clinical texts.
3. **Medication-Effect Mining from Patient Reviews:** Develop NLP model that analyse free-text drug reviews to assess therapeutic effectiveness, prescription appropriateness.
4. **Impact of Sampling Strategies on Model Robustness:** Systematically evaluate random versus balanced sampling—under 75/25 % and 80/20 % train–test splits—to determine their influence on classification reliability and fairness.
5. **Optimisation of pre-processing, vectorisation, and modelling Choices:** Assess the influence of preprocessing (e.g., lemmatization), compare vectorization methods (TF, TF–IDF, BoW, DTM, embeddings), and the evaluation of various classifiers (Logistic Regression, Decision Trees, Random Forests, CNNs, ANNs).

6. **Predictive modelling of drug efficacy:** Establish and evaluate predictive models that predict drug efficacy.
7. **Automatic medical texts classification:** Construct a multi-label classifier that assigns narrative documents to 23 predefined disease categories, analysing the role of vocabulary scope and feature representations.

By pursuing these objectives, the thesis seeks to bridge deep learning and ontology reasoning, delivering more reliable and interpretable methods for large-scale medical text analysis.

Contributions

The thesis advances medical informatics and clinical natural-language processing through the following major contributions:

1. **End-to-End framework for extracting events and temporal concepts:** We introduce a modular deep-learning model (CONSEPTR) to identifie events and temporal expressions in narrative EMRs. Its core architecture combines three complementary convolutional models—standard, transposed, and separable—optimised for hierarchical feature capture.
2. **Temporal-Relation modelling via R_CONSEPTR_8:** To predict temporal links, we design R_CONSEPTR_8, an ensemble model that uses the standard, transposed, separable, and ConLstm convolutional variants and employs majority voting.
3. **Ontology integration with the Jena API:** We couple the DL with domain ontologies (drugs, symptoms, diseases, uber-anatomy, disease drivers, and food materials) using the Java-based Jena API for OWL/RDFS management and SPARQL reasoning. This fusion enriches semantic context.
4. **Ontology integration with Owlready2 library:** A Python-centric alternative is proposed by linking CONSEPTR to the drugs, symptoms, and diseases ontologies through Owlready2 library.
5. **Semantic preservation in patient drug reviews:** We propose an embedding-driven NLP model that retains the nuanced sentiment and context of user-generated drug reviews, enabling reliable prediction of medication efficacy.
6. **Impact of vocabulary scope on vectorisation:** We quantify how local versus global vocabularies influence four classical text representations (TF, TF–IDF, Bag-of-Words, and Document–Term Matrix) and consequently affect the classification accuracy. Through comprehensive experiments, this study investigates the impact of

different text vectorization techniques and vocabulary scopes on the performance of multi-label clinical texts classification models. The evaluation is conducted across 23 distinct disease categories, providing insights into how representation choices influence classification effectiveness.

Collectively, these contributions bridge DL and ontology-based knowledge representation, thereby enhancing the interpretability, robustness, and clinical applicability of medical text-mining systems.

Organization of the Manuscript

The thesis is organized as follows:

- **chapter 1** opens with a characterisation of clinical data, highlighting the complexity of narrative text, before introducing the foundational concepts of ML and DL in a healthcare context. After presenting classical ML models, it contrasts these with DL and, in particular, CNNs. Several CNN variants are detailed, including the Standard, Transposed, Separable, and ConvLSTM architectures. The chapter then pivots to a comprehensive overview of ontologies, formal semantics in OWL, and the interplay between OWL and RDF, highlighting the practical use of the Jena API and Owlready2 library for ontology manipulation, reasoning, and integration.
- **chapter 2** provides a comprehensive review of the most significant research contributions that support the methodological foundations of this thesis. The chapter begins with an overview of key approaches in information extraction, particularly those focused on the recognition of entities and relationships within unstructured clinical narratives. It then explores research related to the automatic classification of medical texts and the detection of drug efficacy from patient-generated content, highlighting the use of ML and DL models in both areas. Following this, the chapter presents a detailed discussion on text vectorization techniques, which are crucial for encoding raw medical text into numerical formats. A dedicated section is also included on natural language preprocessing methods that enhance model performance through linguistic normalization. The chapter concludes with an examination of the role of ontologies in healthcare, emphasizing their contribution to semantic standardization, reasoning, and the enrichment of clinical NLP applications.
- **chapter 3** focuses on enhancing the extraction of clinical events and temporal expressions (TIMEX) by integrating deep learning (DL) models with ontology-based

reasoning. The chapter is structured into three parts. The first part presents the proposed modular DL model in detail. The second part introduces the use of the Jena API for integrating OWL ontologies with DL model, supported by SPARQL querying and reasoning mechanisms. It also explains the Owlready2 library for ontology integration in Python-based environments. Both approaches are evaluated on clinical datasets, with illustrative prediction examples and performance analyses. The final part presents a DL model specifically designed for temporal relation extraction, enabling the prediction of temporal links between entities.

- **chapter 4** addresses the classification of medical texts by exploring the application of neural network models to categorize clinical texts into 23 predefined disease classes. The chapter presents the design of the ANN model and examines how vectorization methods affect performance. Through comparative evaluation.
- **chapter 5** addresses the detection of drug effectiveness using patient reviews. It details the dataset, outlines two text preprocessing strategies (without transformation and with lemmatization), and assesses the influence of sampling techniques. It then introduces traditional and neural network models applied to the task, followed by a comparative analysis of results based on different preprocessing strategies.
- **The General Conclusion** chapter offers a comprehensive synthesis of the thesis by summarizing the key contributions across three major domains: event, temporal expression and temporal link extraction, medical text classification, and drug efficiency prediction. It revisits the main experimental results, and their impact in clinical NLP. The chapter also outlines future research directions and practical applications, highlighting the broader significance and potential of the work.

Chapter 1

Foundations of Machine Learning and Ontologies

Contents

1.1 Introduction 9

1.2 Clinical Data: Nature and Structure 9

1.2.1 Clinical Data 9

1.2.2 Clinical Narrative 10

1.2.3 The Challenges 10

1.3 Core Concepts of Machine Learning 11

1.3.1 Introduction 11

1.3.2 Statistical Learning Models 12

1.3.3 Artificial Neural Networks and Their Deep Variants 17

1.4 Fundamentals of Ontologies 28

1.4.1 Introduction 28

1.4.2 OWL Ontologies 30

1.4.3 Uses in Healthcare 30

1.4.4 Resource Description Framework (RDF) 31

1.4.5 Relationship between RDF and OWL 31

1.4.6 Challenges and Limitations of RDF and OWL Ontologies 32

1.4.7 Jena API 33

1.4.8 Owlready2 Library 35

1.4.9 Usage and Applications 36
1.5 Conclusion 37

1.1 Introduction

This chapter begins by delineating the distinctive characteristics of clinical data, with particular emphasis on the linguistic and structural complexity inherent in narrative medical text. It introduces the core principles of ML and DL. Classical ML mefthods, namely regression techniques, Decision Trees, Random Forests, and k-Nearest Neighbours. Then it focuses on Artificial Neural Networks and, in particular, Convolutional Neural Networks (CNNs). A range of CNN variants—Standard, Transposed, Separable, and ConvLSTM—are examined, each assessed for its capacity to model spatial and temporal patterns in medical narratives. The discussion subsequently pivots to knowledge representation technologies, offering a comprehensive overview of ontologies, the formal semantics of the Web Ontology Language (OWL), and the complementary role of the Resource Description Framework (RDF) in structuring and querying clinical knowledge. Finally, the chapter presents two pivotal ontology engineering toolkits—the Java based Jena API and the Python based Owlready2 library.

1.2 Clinical Data: Nature and Structure

Clinical data provides a critical basis for healthcare, encompassing a wide array of information generated during patient care. This data can range from structured formats, such as numerical values in laboratory results or medication prescriptions, to unstructured formats, including clinical narratives, physician notes, and radiology reports. Understanding the nature, structure, and challenges of clinical data is essential for its effective use in improving healthcare outcomes.

1.2.1 Clinical Data

Clinical data is naturally heterogeneous, reflecting the diversity of healthcare processes and the complexity of patient care. It can be broadly categorized into structured and unstructured data:

- Structured data denotes information that is systematically arranged according to predefined formats. It typically includes demographic details, vital signs, diagnostic codes,

and laboratory test results. Due to its organized nature, structured data is highly suitable for analysis using traditional statistical techniques and ML models [4].

- Unstructured data encompasses free-text clinical narratives, physician notes, discharge summaries, and imaging reports. It represents a substantial portion of clinical information, with estimates suggesting that it accounts for over 80% of data in some healthcare systems. Unlike structured data, unstructured data requires the application of advanced natural language processing (NLP) techniques to extract meaningful insights [5].

1.2.2 Clinical Narrative

Clinical narratives represent an essential source of information within electronic health records (EHRs), offering rich, context-dependent descriptions of patient status, clinical observations, and care plans. Unlike structured data formats, these narratives capture nuanced details that are vital for understanding patient trajectories and the complexity of medical decision-making. However, their unstructured nature introduces considerable variability, as they are often shaped by the individual writing styles of clinicians and the documentation practices specific to each healthcare institution. Furthermore, clinical narratives frequently incorporate abbreviations, ambiguous terms, and context-sensitive expressions, all of which present substantial challenges for automated interpretation and information extraction systems [6].

1.2.3 The Challenges

Despite its potential, clinical data presents unique challenges:

1. Data Volume and Variability: The massive volume of data generated by modern healthcare systems demands scalable solutions for both storage and analysis. Additionally, significant variability in data formats, terminologies, and documentation practices poses challenges for integration and standardization efforts.
2. Ambiguity and Context Dependence: Clinical narratives frequently contain ambiguous terms or abbreviations, which can lead to misunderstanding if context is not carefully considered. For instance, the abbreviation "CP" may refer to "Chest Pain" or "Cerebral Palsy," depending on the clinical context. Misinterpretations of such terms can result in critical errors in diagnosis or treatment [7].
3. Security and Privacy: Confidentiality and data protection are critical concerns in the management of clinical information, which is highly sensitive by nature. Regulatory

frameworks such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States and the General Data Protection Regulation (GDPR) in the European Union establish stringent guidelines to ensure the secure handling and lawful processing of personal health data. Ensuring the secure storage and analysis of such data, while maintaining its utility for research and clinical applications, remains a significant and ongoing challenge [8]. In Algeria, the protection of clinical and personal health data is governed by Law No. 18-07 (2018), which regulates the processing of personal data in line with international data privacy standards. The law is enforced by the Commission Nationale de Protection des Données Personnelles (CNPDP), which oversees the authorization and supervision of sensitive data processing, including health records.

4. Data Quality: Clinical data often contains inconsistencies, missing values, or errors, reflecting the realities of high-pressure clinical environments. Addressing these issues requires robust preprocessing techniques and domain expertise [9].

In conclusion, clinical data, with its diverse nature and structure, provides a rich source of knowledge that supports medical research advancement and enhances the quality of patient care. However, the challenges posed by unstructured data, ambiguity, and privacy concerns necessitate sophisticated computational approaches. Leveraging techniques such as NLP, ontology-based frameworks, and ML can unlock the full potential of clinical data while ensuring its responsible use.

1.3 Core Concepts of Machine Learning

1.3.1 Introduction

Artificial Intelligence (AI) is a broad field that aims to create intelligent systems capable of performing tasks that typically require human intelligence, such as speech recognition, decision-making, and problem-solving [10]. Within AI, Machine Learning (ML) is a subset that focuses on developing algorithms that enable computers to learn from data and make predictions without being explicitly programmed [11].

Deep Learning (DL) is a specialized branch of ML that utilizes multi-layered artificial neural networks to model complex patterns in data [12]. The hierarchy of these concepts can be structured as follows:

- Artificial Intelligence (AI): The broadest field, encompassing all intelligent systems.
- Machine Learning (ML): A subset of AI that focuses on learning patterns from data.

- Deep Learning (DL): A subset of ML that utilizes neural networks with multiple layers.

In this thesis, we employ supervised learning, where algorithms are trained on input-output pairs to learn a mapping function $f(x) = y$. This approach encompasses both regression, which predicts continuous values, and classification, which identifies discrete categories. The following sections provide an overview of the primary methods used.

1.3.2 Statistical Learning Models

As classification models are designed to predict categorical outcomes, this thesis investigates the following approaches:

Logistic Regression: Logistic regression is a widely applied supervised machine learning method used for classification tasks. Unlike linear regression, which predicts continuous numerical values, logistic regression estimates the probability that an input instance belongs to a particular categorical class. There are three primary forms of logistic regression, each suited to a different type of categorical response variable[1]:

1. **Binary logistic regression:** Used when the dependent variable has two possible outcomes, typically coded as 0 and 1. An example would be predicting infection status, such as *infected* (1) versus *not infected* (0).

2. **Multinomial logistic regression:** Applied when the response variable includes more than two unordered categories. For example, classifying disease severity into *no disease* (0), *mild disease* (1), and *severe disease* (2).

3. **Ordinal logistic regression:** Suitable for dependent variables with more than two ordered categories. A typical example is modeling patient satisfaction ratings with hospital services on a scale from 1 (very dissatisfied) to 5 (very satisfied).

Advantages and Limitations: Logistic regression is a widely adopted classification technique, particularly valued in biomedical and clinical domains for its interpretability and efficiency. The model's coefficients offer transparent insights into the influence of predictors on outcome probabilities, supporting informed decision-making. Its computational simplicity and probabilistic outputs further enhance its applicability, especially in binary classification scenarios, with extensions available for multiclass and ordinal outcomes.

However, the method is constrained by certain limitations. It assumes a linear relationship between the independent variables and the log-odds of the dependent variable, which

[1] https://www.geeksforgeeks.org/machine-learning/understanding-logistic-regression/

may not capture complex or non-linear patterns. Its sensitivity to multicollinearity can undermine the stability of coefficient estimates, and it is unsuitable for modeling continuous outcomes.

Decision Tree (DT): DT [13] is a popular supervised learning method used for both classification and regression tasks. They represent a model as a tree-like structure where internal nodes correspond to decision rules, branches represent outcomes, and leaf nodes hold the predicted class or value.

Working Principle: A decision tree recursively partitions the dataset into subsets based on feature values. At each step, the algorithm chooses the best split by selecting a feature and a threshold that maximize the homogeneity of the resulting subsets.

Splitting Criteria: Different impurity measures are used to evaluate how well a split separates the classes in classification trees. The two most common criteria are:

- **Gini Impurity** (I_G): Used in CART (Classification and Regression Trees), it measures the probability of misclassifying a randomly chosen element if it were labeled according to the distribution of the subset.

$$I_G = 1 - \sum_{i=1}^{C} p_i^2 \tag{1.1}$$

  where p_i is the proportion of class i instances in the subset, and C is the number of classes.
- **Entropy** (H): Used in ID3 (Iterative Dichotomiser 3) [14], entropy measures the information gain by reducing uncertainty.

$$H(S) = -\sum_{i=1}^{C} p_i \log_2 p_i \tag{1.2}$$

  where S is the subset being evaluated.

Decision Tree Algorithm: The basic decision tree algorithm follows these steps:

1. Calculate impurity (Gini/Entropy) for the dataset.
2. Evaluate all possible feature splits and compute impurity reduction.
3. Select the best feature and threshold that maximizes impurity reduction.
4. Split the dataset into two or more child nodes.

5. Repeat recursively until a stopping condition is met (e.g., no further gain, max depth, min samples per node).
6. Assign majority class labels to leaf nodes.

Pruning Techniques: To prevent overfitting, decision trees can be pruned:

- Pre-pruning: Stop growing the tree early based on conditions (e.g., max depth, min samples per leaf).
- Post-pruning: Grow the full tree and then remove nodes that do not improve validation accuracy.

Advantages and Limitations: One of the primary strengths of this model lies in its simplicity, making it easy to interpret and visualize. Additionally, it does not require feature scaling or normalization, which simplifies the preprocessing pipeline. The model also handles both categorical and numerical data effectively, allowing for flexibility in diverse datasets.

Despite its strengths, the model is prone to overfitting, particularly when it becomes overly complex; this often necessitates the use of pruning techniques or integration into ensemble methods such as Random Forests. Moreover, the model can be sensitive to small variations in the data, which may result in different decision paths or splits.

Random Forest (RF): RF [15] is an ensemble learning method that combines multiple decision trees to improve predictive accuracy and reduce overfitting. They use a technique called bootstrap aggregation (bagging) to generate diverse trees and aggregate their predictions. RF build multiple decision trees in parallel and aggregate their outputs. Each tree is trained on a random subset of the training data, selected using sampling techniques. Additionally, at each node, a random subset of features is chosen to determine the best split, further enhancing diversity among trees.

Sampling Techniques: RF rely on different sampling methods to create diverse subsets of training data, which helps improve generalization.

- **Bootstrap Sampling (Bagging):** The standard method used in RF, where each tree is trained on a random subset of the training data sampled with replacement. Approximately 37% of the original samples remain unused in each bootstrap sample, forming the out-of-bag (OOB) set for unbiased error estimation [16].
- **Pasting:** Unlike bootstrap sampling, pasting selects random subsets of training data without replacement, ensuring each sample is used at most once in different trees. This method was introduced in [17].

- **Subsampling:** A variation of bootstrap sampling where trees are trained on smaller, fixed-sized subsets of data, improving computational efficiency [18].
- **Stratified Sampling:** Ensures that each class in a classification problem is represented proportionally in the training subsets, reducing bias in imbalanced datasets [19].
- **Bootstrap Weighted Sampling:** Instead of sampling uniformly, instances are sampled based on weights, allowing more difficult samples to be selected more frequently, similar to boosting [20].

RF Algorithm: The Random Forest training process consists of:

1. **Sampling Strategy:** Select a subset of the training data using bootstrap sampling, pasting, or another sampling method.
2. **Tree Growth:** Train a decision tree on each sampled dataset:
   - At each node, select m random features.
   - Find the best split using an impurity measure (e.g., Gini impurity or entropy).
   - Repeat until a stopping criterion (e.g., max depth, min samples per leaf) is met.
3. **Prediction Aggregation:** Combine outputs of all trees:
   - For classification, take a majority vote.
   - For regression, take the average prediction.

Advantages and Limitations of RFs: RFs offer several advantages that make them highly effective for classification and regression tasks. First, they provide high predictive accuracy by reducing the risk of overfitting commonly associated with individual decision trees. They are also well-suited for high-dimensional datasets, effectively handling a large number of input features without requiring feature selection. Moreover, RFs are capable of handling missing data through the use of surrogate splits, making them robust in real-world scenarios with incomplete information. An additional strength lies in their ability to assess feature importance, offering insights into the relative contribution of each variable to the model's predictions [18].

Despite their advantages, RFs tend to be computationally intensive, as training involves constructing and aggregating predictions from a large number of decision trees, which can increase training time and resource usage. Furthermore, the ensemble nature of the model leads to reduced interpretability, making it more challenging to understand decision boundaries compared to a single tree. Lastly, RFs may exhibit bias when applied to highly imbalanced datasets, potentially favoring the majority class unless additional measures, such as

resampling or class weighting, are applied.

K-Nearest Neighbors (KNN): KNN [21] is a non-parametric, instance-based learning algorithm used for classification and regression tasks. It classifies a new sample by finding the k closest data points in the training set and assigning the majority class among them.

Working Principle: KNN operates based on the assumption that similar instances exist in close proximity within a feature space. It uses distance metrics to measure similarity between points and determine the nearest neighbors.

Distance Metrics: The most commonly used distance metrics in KNN are:

- **Euclidean Distance:** The most widely used distance metric, calculated as:

$$d(p, q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{1.3}$$

  where p and q are two feature vectors in an n-dimensional space.
- **Manhattan Distance:** Measures distance as the sum of absolute differences between coordinates:

$$d(p, q) = \sum_{i=1}^{n} |p_i - q_i| \tag{1.4}$$

  This is useful for grid-based applications like city block distances.
- **Minkowski Distance:** A generalization of both Euclidean and Manhattan distances:

$$d(p, q) = \left(\sum_{i=1}^{n} |p_i - q_i|^r \right)^{\frac{1}{r}} \tag{1.5}$$

  where $r = 2$ corresponds to Euclidean distance, and $r = 1$ corresponds to Manhattan distance.
- **Cosine Similarity:** Used for text-based applications, measures the cosine of the angle between two vectors:

$$\text{Cosine Similarity} = \frac{p \cdot q}{||p||||q||} \tag{1.6}$$

  where $p \cdot q$ is the dot product and $||p||, ||q||$ are vector magnitudes.

KNN Algorithm: The KNN algorithm follows these steps:

1. Select the number of neighbors, k.

2. Compute the distance between the test point and all training points.
3. Identify the k nearest neighbors based on the chosen distance metric.
4. For classification: Assign the majority class label among the k neighbors.
5. For regression: Take the average of the target values of the k nearest neighbors.

Advantages and Limitationsof KNN: The KNN algorithm offers several advantages, notably its simplicity and ease of implementation. It performs effectively in multi-class classification tasks and requires no explicit training phase, making it particularly efficient for small datasets. However, KNN also presents notable limitations. Its computational cost can become significant with large datasets due to the need to compute distances between each test instance and all training samples.

1.3.3 Artificial Neural Networks and Their Deep Variants

Artificial neural network consists of interconnected neurons (nodes) arranged in layers. Deep Neural Networks (DNNs) are characterized by the presence of multiple hidden layers, enabling them to capture more abstract and complex features [22]. Several architectures have been developed to handle different types of data:

- **Convolutional Neural Networks (CNNs):** Initially specialized in image processing, they use convolutional layers to capture spatial hierarchies [23].
- **Recurrent Neural Networks (RNNs):** Designed for sequential data, these networks maintain memory of previous inputs, making them suitable for tasks like speech recognition and time-series analysis [24].
- **Transformers:** Introduced in NLP, transformers replace RNNs by using self-attention mechanisms to model long-range dependencies [25].

DL models are trained using backpropagation [26], which adjusts weights in the network based on errors propagated backward. Optimization algorithms like Stochastic Gradient Descent (SGD) and Adam help minimize loss functions and improve model accuracy.

DL has revolutionized a wide range of domains, demonstrating transformative capabilities across various applications. In the field of computer vision, it has enabled significant advancements in tasks such as image classification [27] and object detection [28]. Similarly, in NLP, DL techniques have contributed to breakthroughs in machine translation [29] and sentiment analysis, facilitating more accurate and human-like language understanding. The healthcare sector has also benefited immensely, with DL models being employed for medical diagnosis from imaging data [30] and accelerating processes in drug discovery. Furthermore,

autonomous systems, such as self-driving cars [31] and robotic applications, have leveraged DL to achieve higher levels of decision-making autonomy and real-time adaptability.

Despite the notable success of Deep Learning in various domains, it continues to face significant challenges. One major limitation is the high computational cost associated with training large-scale models, which necessitates the use of powerful hardware resources such as Graphics Processing Units (GPUs) or Tensor Processing Units (TPUs). This requirement can pose accessibility barriers for researchers and institutions with limited computational infrastructure. Another critical challenge is data dependency, as the performance of DL models heavily relies on the availability of large, well-annotated datasets. In many specialized fields, particularly in the medical domain, obtaining sufficiently large and accurately labeled datasets can be difficult, which may hinder the effective application of DL techniques.

Artificial Neural Networks (ANNs): ANNs are computational models inspired by the biological neural networks in the human brain. They consist of interconnected nodes (neurons) arranged in layers and are widely used for pattern recognition, classification, and regression tasks. ANN consist of three main types of layers:

- **Input Layer:** Accepts raw input features.
- **Hidden Layers:** Perform computations and extract representations.
- **Output Layer:** Produces the final prediction.

A feedforward neural network (FNN) processes inputs in a one-way fashion from the input layer to the output layer.

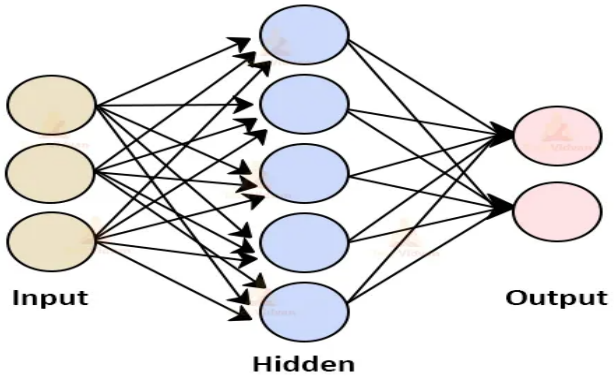


Figure 1.1: Basic architecture of an Artificial Neural Network.

Perceptron Model: ANN are based on the perceptron model introduced by [32]. The perceptron is the simplest type of artificial neuron network (Figure 1.2), defined by Eq. 1.7

:

$$y = f\left(\sum_{i=1}^{a} w_i x_i + b\right) \tag{1.7}$$

where:

- x_i are input features.
- w_i are weights assigned to each input.
- b is the bias term.
- f is the activation function.

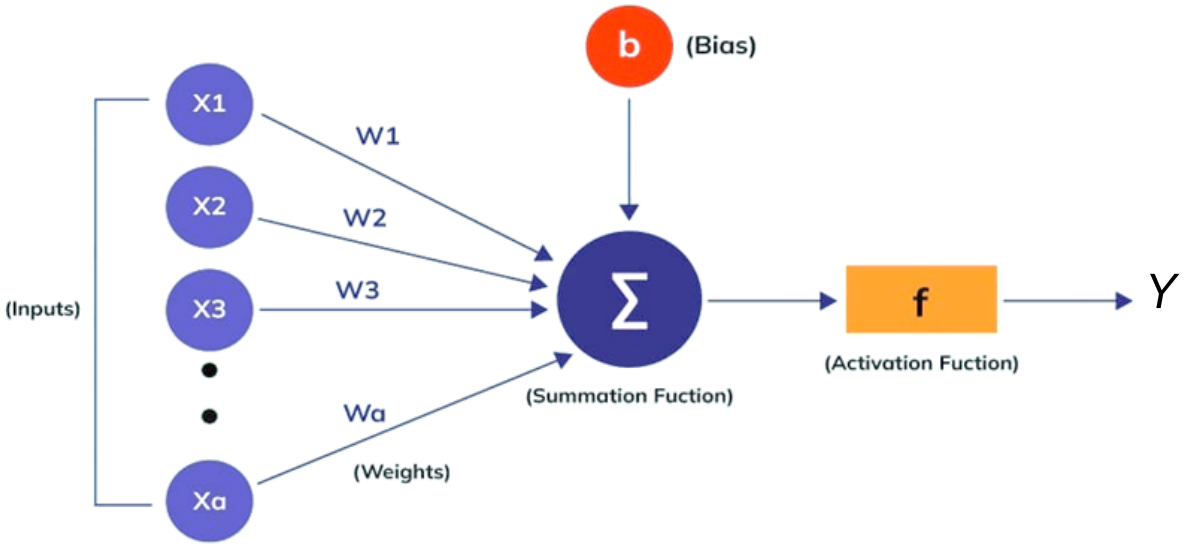


Figure 1.2: Perceptron architecture.

Activation Functions: Activation functions introduce non-linearity in ANNs, enabling them to model complex patterns. Common activation functions include:

- **Sigmoid:**

$$f(x) = \frac{1}{1 + e^{-x}} \tag{1.8}$$

  Used for binary classification but suffers from the vanishing gradient problem.
- **Hyperbolic Tangent (Tanh):**

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{1.9}$$

  Outputs values in the range $[-1, 1]$, reducing the bias in activations.

- **ReLU (Rectified Linear Unit):**

$$f(x) = \max(0, x) \tag{1.10}$$

  ReLU is computationally efficient and helps mitigate vanishing gradients.
- **Leaky ReLU:**

$$f(x) = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases} \tag{1.11}$$

  A variant of ReLU that prevents dying neurons by allowing small gradients for negative inputs.

Backpropagation Algorithm: Backpropagation, introduced by [33], is a method used to update ANN weights efficiently. It consists of two phases:

1. **Forward Pass:** Compute output predictions using the current weights.
2. **Backward Pass:** Adjust weights using gradient descent to minimize the error.
   The error is computed using the loss function, such as Mean Squared Error (MSE) for regression:

$$L = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{1.12}$$

   or Cross-Entropy Loss for classification:

$$L = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{1.13}$$

   The weight update follows:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \eta \frac{\partial L}{\partial w_{ij}} \tag{1.14}$$

   where η is the learning rate.

Optimization Techniques: Gradient descent variations improve training efficiency:

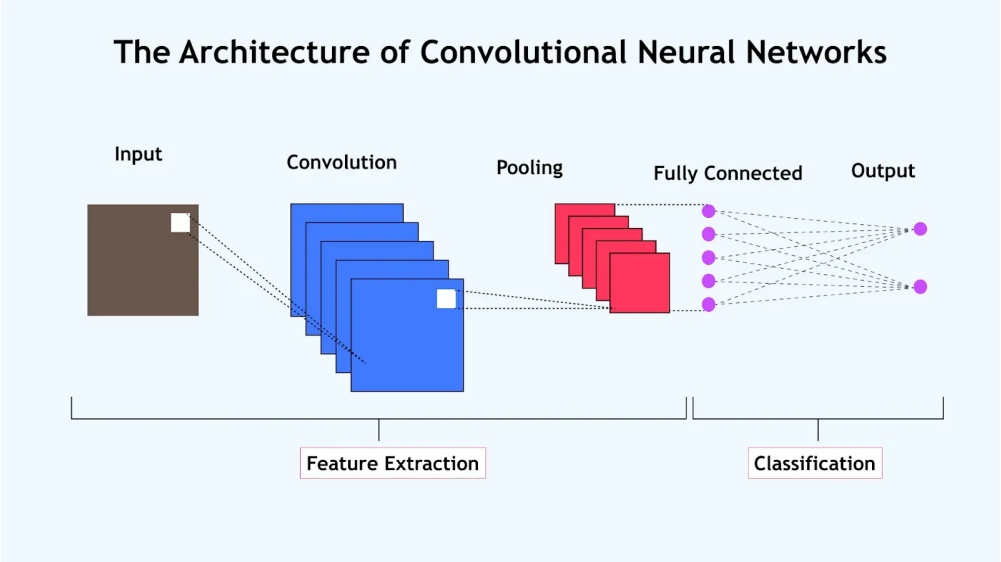
- **Stochastic Gradient Descent (SGD)**: Updates weights after each sample.
- **Mini-Batch Gradient Descent**: Updates weights after processing a small batch.
- **Adam (Adaptive Moment Estimation)** [34]: Combines momentum and adaptive learning rates.

Advantages and Limitations of ANN: ANN offer significant advantages in modeling complex, non-linear relationships within data. They are particularly well-suited for tasks such as image recognition, speech processing, and NLP. Additionally, ANN are capable of handling large-scale datasets containing numerous features, making them highly effective in domains requiring the analysis of high-dimensional information. Despite their strengths, ANNs also present several limitations. They are computationally expensive, often requiring access to high-performance GPUs to efficiently train large models. Furthermore, ANNs are prone to overfitting, particularly when dealing with limited training data, necessitating the use of regularization techniques such as dropout to improve generalization and model robustness.

Convolutional Neural Networks (CNN): CNNs, first introduced by [35], are a class of DL models designed for image recognition and feature extraction. They are particularly effective in handling spatially structured data such as images and videos. CNNs consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers, and they rely conconvolution operations to extract features automatically from input data.

CNN Architecture: As illustrated in Figure **??**, a typical CNN consists of the following layers:

- **Convolutional Layer:** Extracts spatial features from input data.
- **Activation Function:** Introduces non-linearity (e.g., ReLU).
- **Pooling Layer:** Reduces spatial dimensions while preserving essential information.
- **Fully Connected Layer:** Maps extracted features to final predictions.



Basic Convolutional Neural Network architecture]Basic Convolutional Neural Network architecture.

Convolutional Layer: The core operation in CNNs is the convolution operation, defined mathematically as:

$$S(i,j) = (X * W)(i,j) = \sum_m \sum_n X(i+m, j+n) W(m,n) \tag{1.15}$$

where:

- X is the input image (or feature map).
- W is the convolution filter (kernel).
- $S(i,j)$ is the output feature map.

This operation slides the filter over the input image, extracting local features such as edges, textures, and patterns.

Activation Function: To introducenon-linearity, CNNs commonly use ReLU, defined in Eq. 1.10. ReLU accelerates training and helps prevent the vanishing gradient problem.

Pooling Layer: The pooling layer performs a downsampling, reducing the spatial dimensions while preserving the most relevant features. Common pooling methods include:

- **Max Pooling:** Selects the maximum value in a window.

$$P(i,j) = \max_{m,n} X(i+m, j+n) \tag{1.16}$$

- **Average Pooling:** Computes the average of values in a window.

Fully Connected Layer: Fully connected layers receive flattened feature maps and produce the final classification or regression output. The output is computed as:

$$y = f(Wx + b) \tag{1.17}$$

where W is the weight matrix, x is the input feature vector, and b is the bias term.

Types of CNN Architectures: CNN architectures have evolved significantly, incorporating advanced techniques to improve performance.

Standard CNN: The standard convolution operation proceeds in a stepwise manner Figure 1.3. Initially, a kernel is applied to a localized region of the input, producing a single output value. Subsequently, the kernel is shifted—typically by one position when the stride is 1, and the operation is repeated. This process continues until the full set of output features is generated.

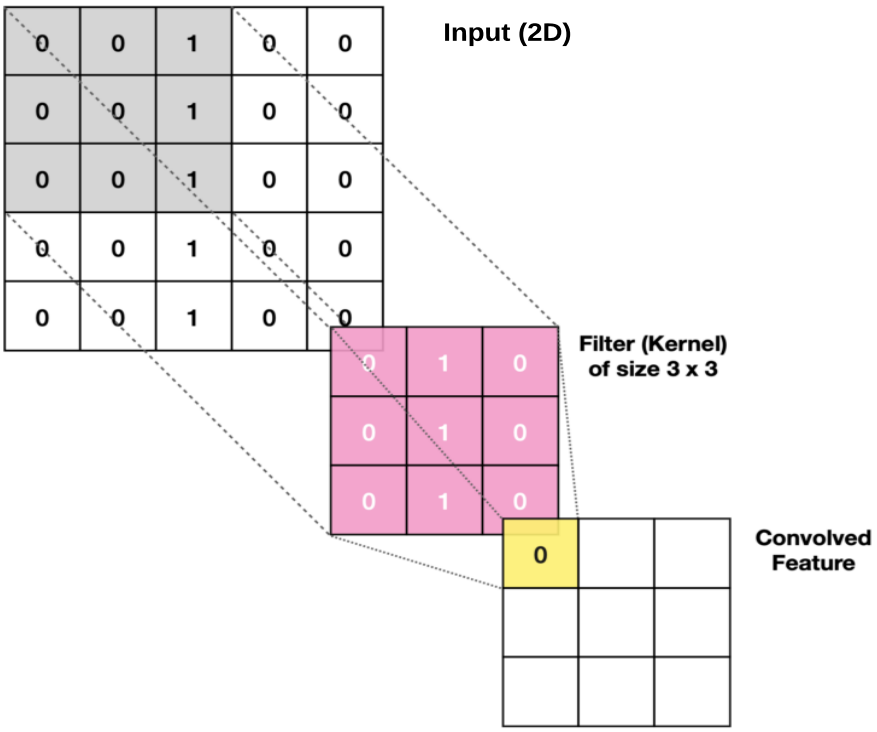


Figure 1.3: The foundational mechanism of standard CNN, stride=1.

Transposed CNN: A transposed CNN (also called a deconvolutional) performs up-sampling operations that increase the spatial resolution of feature maps, as opposed to the down-sampling performed by standard convolutional layers. It is commonly used for image generation, segmentation, and super-resolution [36]. Figure 1.4 illustrates the fundamental concept of transposed CNNs. In this example, a 2×2 input feature map is transformed into a 4×4 output using a 2×2 kernel and a stride of 2.

The process begins by taking each individual element from the input matrix and performing an element-wise multiplication with the convolution filter. Each resulting intermediate product is temporarily stored in a separate output space. This operation is repeated for all entries in the input matrix. Once all positions have been processed, the overlapping values in the intermediate outputs are summed to form the final 4×4 output feature map.

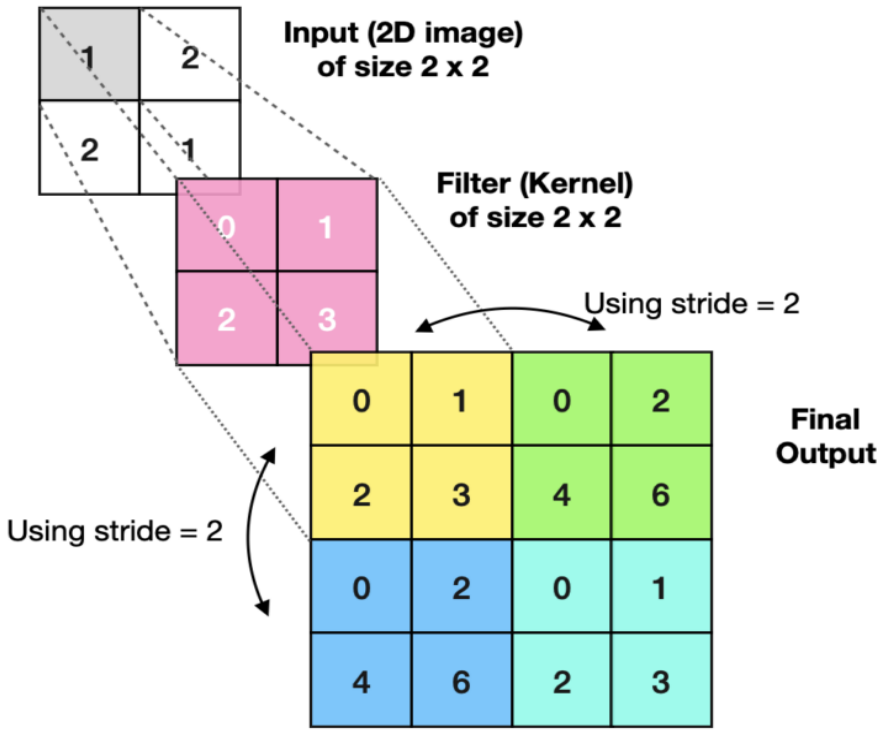


Figure 1.4: The foundational mechanism of transposed CNN.

Separable CNN: Separable CNNs replace standard convolutions with depthwise separable convolutions, which decompose the operation into:

- **Depthwise Convolution:** Applies a single filter per input channel.
- **Pointwise Convolution:** Uses a 1×1 convolution to combine channels.

This significantly reduces computational cost, as used in MobileNet [37]. In the conventional

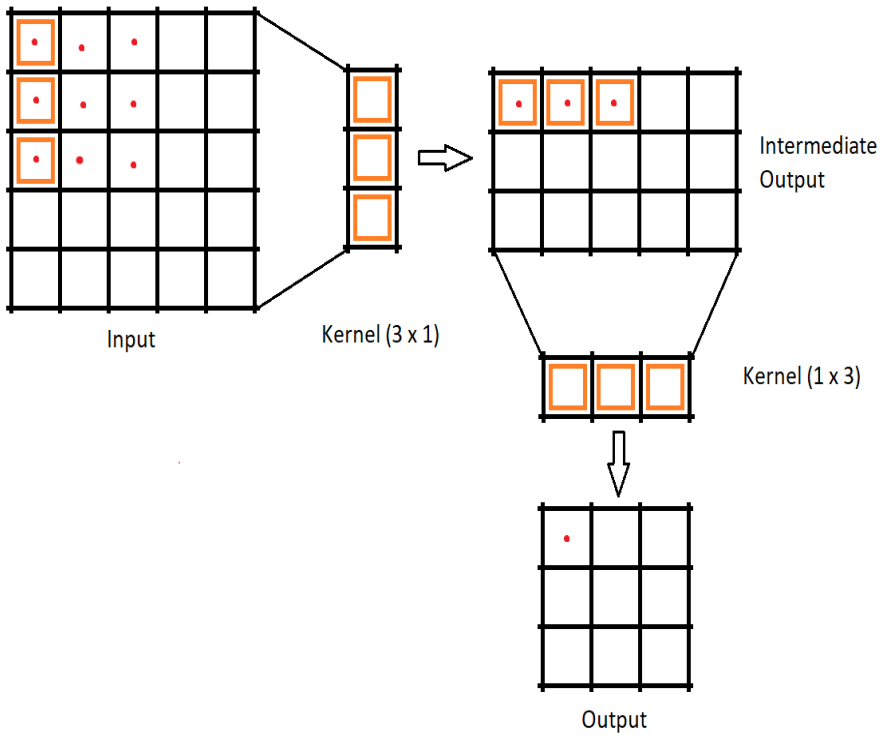


Figure 1.5: The basic mechanism of seperable CNN, stride=1.

convolutions, if the kernel is 3 x 3 then the number of parameters would be 9. In spatially separable convolution we divide the kernel into two kernels of shapes 3 x 1 and 1 x 3 (Figure 1.5). The input is first convolved with 3 x 1 kernel and then with 1 x 3, then the number of parameters would be $3 + 3 = 6$. So less matrix multiplication is required. In spatially separable convolution, a 2D kernel of size 3×3 can be approximated by the outer product of two 1D kernels—a vertical kernel of size 3×1 and a horizontal kernel of size 1×3. This approach significantly reduces the number of parameters and computational cost.

Suppose the original 3×3 kernel is:

$$K = \begin{bmatrix} a & b & c \\ d & e & f \\ z & s & r \end{bmatrix}$$

In a separable convolution, we approximate K using the outer product of two vectors:

$$K \approx \begin{bmatrix} u \\ v \\ w \end{bmatrix} \cdot \begin{bmatrix} x & y & z \end{bmatrix} = \begin{bmatrix} ux & uy & uz \\ vx & vy & vz \\ wx & wy & wz \end{bmatrix}$$

This approximation holds exactly only if the original kernel K is of rank 1. Otherwise,

it is an approximation often obtained during training.

For example:

$$\begin{bmatrix}1\\1\\1\end{bmatrix} \cdot \begin{bmatrix}+1 & 0 & -1\end{bmatrix} = \begin{bmatrix}+1 & 0 & -1\\+1 & 0 & -1\\+1 & 0 & -1\end{bmatrix}$$

Convolutional LSTM (ConvLSTM): Traditional Fully Connected LSTM (FC-LSTM) models present a significant limitation in handling spatiotemporal data: they require the input to be flattened into one-dimensional vectors, which results in the loss of essential spatial relationships inherent in structured data. To overcome this challenge, the ConvLSTM model introduces an architectural modification where all key components (including inputs $X_1, \ldots, X_t$, cell states $C_1, \ldots, C_t$, hidden states $H_1, \ldots, H_t$, and gating functions i_t, f_t, o_t) are represented as 3D tensors. In these tensors, the last two dimensions correspond to the spatial layout (i.e., rows and columns), allowing the network to preserve spatial structure throughout the temporal sequence [38].

Each state and input tensor can be interpreted as a feature map distributed over a spatial grid. The future state of any grid cell is influenced not only by its own history but also by the states of its spatial neighbors. This is made possible by applying convolution operations in both the input-to-state and state-to-state transitions, as opposed to the fully connected transformations used in traditional LSTMs.

The ConvLSTM equations are defined as follows, where '$*$' denotes the convolution operator, '$\circ$' represents the Hadamard (element-wise) product, and σ is the sigmoid activation function:

$$\begin{aligned}
i_t &= \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \circ C_{t-1} + b_i)\\
f_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} \circ C_{t-1} + b_f)\\
C_t &= f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c)\\
o_t &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} \circ C_t + b_o)\\
H_t &= o_t \circ \tanh(C_t)
\end{aligned}$$

This structure (Figure 1.6) enables the ConvLSTM to simultaneously capture temporal dependencies and preserve spatial patterns, making it highly effective for tasks involving spatiotemporal dynamics such as video prediction, weather modeling, and medical imaging analysis.

The ConvLSTM layer in Keras integrates the temporal processing capabilities of Long Short-Term Memory (LSTM) networks with the spatial feature extraction strengths of CNNs. Unlike traditional LSTM layers that utilize fully connected operations, ConvLSTM replaces

these with convolutional operations, enabling the model to capture both spatial and temporal dependencies in sequential data[2]. This method has proven especially effective in domains

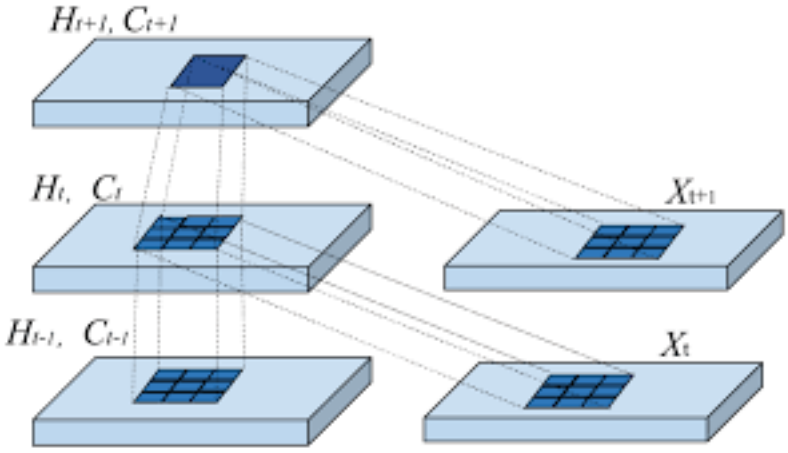


Figure 1.6: ConvLSTM's structure internal.

such as video processing, speech recognition, and time-series prediction in finance [39].

Advantages and Limitations of CNNs: CNNs offer several notable advantages in the context of pattern recognition and computer vision tasks. They are capable of automatically learning hierarchical spatial features from raw input data, reducing the need for manual feature engineering. CNNs also require significantly fewer parameters compared to fully connected neural networks, making them more efficient in terms of memory usage. Moreover, they have demonstrated state-of-the-art performance across a variety of image recognition tasks. However, CNNs are not without limitations. Their training is computationally intensive and typically necessitates high-performance GPUs for practical use. Additionally, CNNs require large labeled datasets to achieve good generalization, which may not always be available in specialized domains.

In conclusion, Machine Learning has evolved into a vast field with numerous applications. Supervised learning is used for prediction tasks, unsupervised learning for pattern discovery, and DL continues to advance ML capabilities in areas like image recognition and NLP. CNNs, in particular, have played a crucial role in computer vision, medical imaging, and speech processing.

[2] https://keras.io/api/layers/recurrent_layers/conv_lstm1d/

1.4 Fundamentals of Ontologies

1.4.1 Introduction

Ontology, in the context of computer science and artificial intelligence, is a structured and formal representation of knowledge. It defines concepts, relationships, and rules within a specific domain, enabling both machines and humans to share and understand information efficiently. The role of ontologies has grown significantly, influencing fields such as artificial intelligence, semantic web technologies, and biomedical informatics.
Ontologies have multiple definitions depending on the discipline:

- **Philosophical Definition:** Ontology is a branch of metaphysics that explores the nature of existence, reality, and being [40].
- **Computer Science Definition:** Gruber [41] defined an ontology as an "explicit specification of a conceptualization."
- **Knowledge Engineering Definition:** A formal and structured model that describes concepts, entities, attributes, and relationships within a domain [42].
- **Semantic Web Definition:** A set of defined terms and relationships that facilitate interoperability among diverse data sources on the web [43].

The development of ontologies has progressed through several distinct phases, evolving from philosophical roots to foundational components in modern artificial intelligence and semantic technologies.

Early Stages (Pre-1990s): In its earliest form, the concept of ontology was deeply rooted in philosophy and linguistics, where it was primarily concerned with the nature of being and categorization. During this period, ontological principles began to influence database systems, helping to define structured data models for organizing and retrieving information [44].

Development Phase (1990s): The 1990s marked a turning point as ontologies gained prominence in the fields of AI and knowledge representation. Notably, Gruber's work [41] laid the foundation for ontology engineering as a formal discipline within AI. This era also saw early efforts toward standardization, with the introduction of languages such as the Knowledge Interchange Format (KIF) and Ontolingua, which enabled shared and reusable conceptualizations [45].

Maturity Phase (2000s – Present): With the rise of the semantic web, ontologies assumed a central role in enabling machines to interpret and reason over structured data [43]. This phase witnessed the development of key ontology languages, including RDF (Resource Description Framework), OWL (Web Ontology Language), and SPARQL for querying semantic data [46]. Ontologies have since been widely adopted in domains such as Big Data, ML, and Biomedical Informatics [47], and now form a critical backbone in technologies like Linked Data and Knowledge Graphs [48].

Ontology Architectures: Ontology architectures define how knowledge is structured and utilized. Common architectures include:

- **Monolithic Ontology:** A single large ontology covering all concepts in a domain (e.g., WordNet [49]).
- **Modular Ontology:** Divides the domain into smaller, interconnected ontologies, enabling reusability and scalability (e.g., Gene Ontology [47]).
- **Distributed Ontology:** Multiple ontologies linked through semantic mappings, used in large-scale systems (e.g., Linked Open Data (LOD) Cloud[3]).
- **Upper Ontology:** A high-level ontology that provides a common framework for lower-level ontologies (e.g., DOLCE, SUMO [50]).

Advantages and Disadvantages of Ontologies: Ontologies provide several benefits, particularly in AI and knowledge management:

- **Knowledge Sharing & Reusability:** Facilitates interoperability between systems [42].
- **Semantic Interoperability:** Ensures a common understanding among heterogeneous systems, particularly in semantic web and data integration [43].
- **Improved Search & Information Retrieval:** Knowledge graphs in search engines enhance search accuracy.
- **Logical Reasoning & Inference:** AI systems can infer new knowledge from existing facts [51].
- **Scalability & Extensibility:** New knowledge can be integrated without requiring system-wide modifications.

Despite their advantages, ontologies face several challenges:

[3] https://www.omg.org/dol/

- **High Development Cost & Complexity:** Requires expertise in ontology engineering and formal logic.
- **Computational Overhead:** Large ontologies demand high computational resources for reasoning [51].
- **Ontology Evolution Challenges:** Updating ontologies without compromising consistency is difficult [52].

In conclusion, ontologies serve as a fundamental component of knowledge representation, AI, and semantic web technologies. Where they provide structured knowledge management.

1.4.2 OWL Ontologies

Web Ontology Language (OWL) ontologies are formalized vocabularies that allow for the specification of a shared understanding of a domain among a community of users. They play a critical role in knowledge representation on the Web, enabling effective communication and interoperability among systems by defining classes, properties, and the relationships between them. The OWL, first introduced in 2004, has since evolved into OWL 2, which provides enhanced features and better support for ontology development and sharing across various applications[4].

OWL 2 ontologies are stored as semantic web documents and include classes, properties, individuals, and data values. These ontologies are primarily exchanged as RDF documents and can be used in conjunction with information articulated in RDF [53]. The formal semantics of OWL specifies how to derive logical consequences from ontologies, enabling machine interpretability of the content defined within them[5]. The OWL 2 Document Overview serves as an introduction to its new features and outlines the differences between OWL and its successor, OWL 2, while detailing the theoretical and implementation rationale behind its design[53]. OWL facilitates greater machine understanding of web content, thereby enhancing the semantic capabilities of web applications and data integration processes.

1.4.3 Uses in Healthcare

In healthcare, OWL ontologies have been leveraged to create comprehensive frameworks for representing medical knowledge, such as clinical guidelines, terminologies, and patient records. Ontology like the Open Biological and Biomedical Ontology (OBO) Foundry provides standardized vocabularies that facilitate data exchange and interoperability among

[4] https://www.w3.org/TR/2012/REC-owl2-overview-20121211/
[5] https://medium.com/understanding-the-web-ontology-language-owl-a-deep-dive-a46abbcafec

healthcare applications. By enabling precise semantic descriptions of medical concepts and their interrelations, OWL ontologies enhance decision support systems, improve data sharing, and enable the development of applications that can reason over medical knowledge.

Narrative medical texts, which encompass patient histories, clinical notes, and medical literature, can benefit from the integration of OWL[6] ontologies by providing a structured representation of the information contained within these texts. This allows for better indexing, retrieval, and analysis of narrative content, thus improving the overall quality of healthcare delivery. The use of ontologies can also assist in the extraction of key insights from unstructured data, enabling healthcare professionals to make more informed decisions based on comprehensive analyses of patient information and relevant medical knowledge.

1.4.4 Resource Description Framework (RDF)

The RDF is a foundational standard for representing structured and semantically linked data on the Web. Developed by the World Wide Web Consortium (W3C) in 1999, RDF was introduced as a core component of the semantic web [54]. At its core, RDF models data as triples consisting of a subject, predicate, and object, forming a directed labeled graph that supports flexible and extensible representation of information [55].

Each element within an RDF triple is typically identified using Uniform Resource Identifiers (URIs), ensuring global uniqueness and disambiguation. Objects in RDF triples may also be literal values, such as strings or numbers [56]. RDF supports multiple serialization formats to represent its data model, including RDF/XML, Turtle (Terse RDF Triple Language), and JSON-LD, each suited for different use cases and system architectures[7].

In practice, RDF plays a central role in a variety of domains, including data integration, knowledge representation, and artificial intelligence. Its graph-based structure allows for merging and querying data across diverse ontologies and schemas, thereby enabling semantic interoperability between distributed systems [57]. RDF has been applied in numerous real-world scenarios, such as linking product catalogs with consumer reviews and merchant data, or integrating biomedical knowledge bases across different institutions.

1.4.5 Relationship between RDF and OWL

RDF and the OWL are two core technologies that underpin the semantic web. RDF serves as a foundational framework for representing structured data using subject-predicate-object

[6] https://www.w3.org/TR/owl2-overview/
[7] https://www.techtarget.com/definition/RDF

triples, allowing for flexible and interoperable data interchange across heterogeneous systems [55]. OWL builds upon RDF by introducing richer ontological constructs and formal semantics based on description logics, thereby enabling automated reasoning over complex domain knowledge [54, 58]. This enhanced expressiveness allows OWL to define class hierarchies, property characteristics (e.g., transitivity, symmetry), restrictions, and axioms, facilitating more precise modeling of relationships and constraints within data.

Together, RDF and OWL provide a layered approach to semantic representation, where RDF forms the syntactic backbone and OWL contributes the semantic depth required for intelligent applications. Their integration is crucial for knowledge-based systems, enabling tasks such as consistency checking, classification, and inferencing. These technologies are especially important in areas such as biomedical informatics, intelligent agents, and data integration on the Web.

1.4.6 Challenges and Limitations of RDF and OWL Ontologies

Implementing RDF and OWL ontologies in data management and semantic web applications introduces a range of challenges that can hinder their widespread adoption and effectiveness.

- **Complexity of Technologies:** Mastering RDF and OWL requires a solid understanding of formal logic, graph theory, and ontology design. This steep learning curve can discourage developers and domain experts from effectively adopting these technologies [54, 56].
- **Data Quality and Availability:** The performance of semantic systems using RDF and OWL is highly dependent on the availability of well-structured and high-quality data. Issues such as inconsistency, incompleteness, and data sparsity can significantly impair their effectiveness [59].
- **Performance Issues:** Querying large RDF graphs using SPARQL can lead to high computational costs. Performance bottlenecks often arise with large-scale datasets, especially in real-time or complex querying scenarios [60].
- **Interoperability Challenges:** Diverse organizations may use different vocabularies or ontologies making semantic interoperability difficult. Aligning heterogeneous data models is a non-trivial task [57].

These challenges must be addressed to ensure the effective deployment of RDF and OWL technologies in complex, real-world applications.

In conclusion, through its robust architecture, RDF not only supports the representation of data but also serves as a critical underpinning for more advanced semantic technolo-

gies, including OWL ontologies, which further extend its functionalities for modeling more intricate relationships and constraints in knowledge representation.

1.4.7 Jena API

The Jena API is a robust framework developed by the Apache Software Foundation for building semantic web applications that manipulate and query RDF data. It supports multiple ontology languages, including RDFS (RDF Schema) and OWL, enabling the creation of sophisticated data models that reflect complex relationships[8].

Notable features include RDF serialization and deserialization, SPARQL query optimization, reasoning capabilities, and integration with Linked Data Platform (LDP) containers. Additionally, Jena facilitates multi-threading for performance enhancement and provides modular ontology structures to streamline complex data management.

Principal Features of the Jena API: Jena API offers an extensive suite of features tailored for RDF data manipulation. One of its fundamental capabilities is the serialization and deserialization of RDF data in multiple syntaxes, including Turtle, RDF/XML, and N-Triples, which facilitates interoperability and standardized data exchange across platforms[9]. Serialization in these formats is essential for aligning with W3C recommendations and ensuring compatibility in distributed systems, as noted in Heath and Bizer's work on Linked Data [61].

Jena also supports advanced SPARQL query execution through its ARQ engine, employing optimization strategies such as optimizer hints and caching mechanisms to reduce computational overhead and improve performance in large-scale data environments[10]. The importance of query optimization in semantic databases is further discussed by Hitzler et al. [54], who emphasize its role in scalable semantic reasoning.

Jena's inference subsystem integrates rule-based and OWL-based reasoners, enabling automatic logical inference and enrichment of RDF graphs through ontology-driven semantics[11]. This aligns with concepts discussed in Antoniou and van Harmelen's book on semantic web ontologies [55], where reasoning is presented as a cornerstone of intelligent data representation.

Additionally, Jena incorporates support for the Linked Data Platform (LDP), allowing RDF resources to be managed via container models, promoting modularity and semantic clar-

[8] https://jena.apache.org/
[9] https://jena.apache.org/documentation/io/
[10] https://jena.apache.org/documentation/query/
[11] https://jena.apache.org/documentation/inference/

ity in data organization[12]. Performance scalability is enhanced through built-in support for multi-threading and parallel query processing, making Jena well-suited for high-throughput and real-time semantic.

Finally, Jena offers prefix mapping and serialization control features that streamline RDF output by abbreviating long URIs and allowing tailored serialization schemas[13]. These capabilities collectively position Jena as a foundational toolkit for semantic web development and data integration across heterogeneous sources, as supported in the broader literature on semantic technologies.

Applications of Jena API in Semantic Web Development: Jena API serves as a comprehensive framework for constructing semantic web and linked data applications. Its versatile architecture supports various facets of semantic data management, including vocabulary development, API design, and ontology handling.

- **Business Domain Vocabulary Development:** Jena facilitates the creation of structured vocabularies within business domains by enabling the modeling of RDF graphs that represent domain-specific concepts and relationships. This structured approach ensures consistency across APIs and enhances interoperability among disparate systems. The use of standardized vocabularies aligns with best practices in semantic data modeling, promoting shared understanding and reuse of data structures [61].
- **API Architectural Standards Compliance:** In the realm of API development, Jena supports adherence to architectural standards such as RESTful principles [62], security protocols, and versioning strategies. Its integration capabilities allow developers to construct APIs that are scalable and maintainable, aligning with modern software engineering practices.
- **Ontology Management and Reasoning:** Jena's ontology API provides robust support for the creation, modification, and integration of ontologies, particularly those conforming to OWL2 specifications. This functionality is crucial for applications requiring complex knowledge representation and reasoning capabilities. By enabling logical inference over RDF datasets, Jena facilitates the derivation of implicit knowledge, thereby enriching data semantics and supporting intelligent decision-making processes [54].

[12] https://jena.apache.org/documentation/ldp/
[13] https://jena.apache.org/documentation/io/

Challenges and Limitations: Jena's performance may degrade with large RDF datasets due to memory constraints, necessitating database-backed solutions like TDB2[14]. In addition, security considerations, including data sanitization and access control, are critical to mitigating potential vulnerabilities in RDF data.

In conclusion, Jena remains a leading API for semantic web applications, providing extensive ontology management and reasoning capabilities. Despite its limitations, its robust feature set and wide adoption make it an invaluable tool for developers.

1.4.8 Owlready2 Library

Owlready2 is a Python module designed for ontology-oriented programming, facilitating the manipulation and reasoning of OWL 2.0 ontologies within Python environments. Owlready2 enables seamless integration of ontological concepts as Python objects, allowing for efficient ontology management and reasoning tasks. A distinguishing feature of Owlready2 is its support for reasoning capabilities. This allows for automatic classification and inference within ontologies, which is particularly beneficial in domains requiring complex hierarchical structures and relationships, such as biomedical informatics [63].

The library's architecture is optimized for performance and memory efficiency, utilizing an SQLite3-based quadstore to handle large ontologies effectively. This design choice ensures that Owlready2 can manage extensive datasets without significant degradation in performance[15].

Furthermore, Owlready2 supports the import and export of ontologies in multiple formats, including RDF/XML, OWL/XML, and N-Triples, enhancing its interoperability with other tools and platforms. Its ability to integrate with collaborative tools and version control systems aligns with the growing emphasis on decentralized and collaborative knowledge management in contemporary research and development environments [64]. Overall, Owlready2 serves as a robust tool for ontology management in Python.

Key Functionalities of Owlready2: Owlready2 provides a comprehensive and flexible framework for ontology handling and knowledge representation, making it a powerful tool for both researchers and developers working in the field of semantic technologies. One of its core strengths lies in its ability to efficiently parse and manipulate ontology files, particularly those encoded in RDF and OWL formats. This functionality enables users to extract and programmatically interact with essential ontological elements such as classes,

[14] https://jena.apache.org/documentation/tdb/
[15] https://owlready2.readthedocs.io/en/latest/onto.html

object properties, and data properties, thus supporting a wide range of ontology-driven applications.

A notable feature of Owlready2 is its integration with graph databases like Neo4j. This capability allows ontology entities to be mapped directly onto graph structures, where classes become nodes and relationships are expressed through Cypher queries. Such integration facilitates advanced graph-based reasoning and visualization of ontological knowledge. Moreover, Owlready2 adopts a Pythonic API that simplifies ontology manipulation, enabling intuitive and rapid development of ontology-based systems. This design is particularly beneficial for integrating ontological reasoning into artificial intelligence workflows.

The library is also designed for cross-platform compatibility, ensuring reliable deployment across different operating systems and development environments. Beyond basic ontology editing, Owlready2 incorporates robust reasoning capabilities through its support for standard description logic reasoners, such as Pellet and Hermit. These reasoners enable logical inference over ontologies, allowing for the detection of implicit relationships and the validation of semantic consistency within complex knowledge models.

Additionally, Owlready2 supports a variety of semantic data formats, including RDF, which promotes interoperability and efficient data exchange across distributed linked data systems. Its ability to interoperate with other Python libraries further enhances its utility. For example, it can be used alongside libraries for semantic reasoning, knowledge graph construction, and data analytics to build advanced, ontology-aware applications[16].

1.4.9 Usage and Applications

Owlready2 allows for the definition of classes, properties, and instances, making it a valuable tool in fields such as biomedical research, AI, and data integration. It enables the hierarchical definition of ontological structures, ensuring logical consistency. The library also provides methods for creating, deleting, and modifying ontology instances, thereby facilitating efficient knowledge representation and semantic reasoning.

The ability to integrate with reasoning engines like Pellet and Hermit, as well as ontology management tools such as Protégé, makes it highly useful for ontology-based verification and validation. Below are some of its key application areas:

- **Data Integration and Interoperability:** Owlready2 provides a standardized vocabulary to facilitate data integration across heterogeneous sources, ensuring semantic interoperability [63].

[16] https://github.com/Delabeye/wonka

- **Semantic Searching and Retrieval:** Enhances search capabilities through semantic querying, improving knowledge discovery in structured and unstructured datasets [65].
- **Biomedical Applications**[17]**:** Aids in clinical decision support, disease diagnosis, and drug discovery by structuring medical knowledge in an ontological framework.

In conclusion, Owlready2 has established itself as a powerful and flexible ontology management library, benefiting from a growing community of researchers and developers. With continuous improvements in reasoning capabilities, data integration, and semantic querying, it plays a crucial role in advancing ontology-driven AI.

1.5 Conclusion

In summary, this chapter has laid the conceptual and methodological groundwork necessary for tackling complex clinical information extraction tasks. It has explored two complementary dimensions: data-driven learning and semantic knowledge representation. On the one hand, we examined how classical and DL models—including CNN variants—can uncover patterns in narrative medical texts, particularly when applied to temporally rich and semantically diverse data. On the other hand, we established the relevance of ontologies and semantic frameworks such as OWL and RDF in enriching and formalizing medical knowledge, particularly through tools like Jena API and Owlready2, which bridge ML and structured reasoning. What emerges from this dual focus is a recognition that high-performance clinical NLP systems must not only learn from data but also reason with knowledge. This integration of DL and symbolic approaches is essential for enabling nuanced understanding, robust generalization, and interpretability in health informatics applications. The insights gained here serve as a foundation for the subsequent chapters, where these techniques are operationalized in real-world tasks such as event, temporal expression, and temporal relation extraction, medical text classification, and drug effectiveness analysis.

[17] https://owlready2.readthedocs.io/en/v0.42/

Chapter 2

State of the Art of Themes Extraction

Contents

2.1 Introduction 38

2.2 Information Extraction 39

2.3 Events, Temporal Expression, and TLink 40

2.4 Detection of Drugs Efficiency 43

2.5 Clinical Documents Classification 44

2.6 Text Vectorization 45

2.7 Natural Language Preprocessing 46

2.8 Ontologies in healthcare 47

2.9 Conclusion 48

2.1 Introduction

This chapter is dedicated to a comprehensive review of the relevant literature that underpins the methodological framework of this thesis. The chapter systematically examines existing work in clinical NLP, with a focus on extracting structured knowledge from unstructured medical narratives. Given the complexity and diversity of clinical text [66], a review of foundational and state-of-the-art techniques is essential for contextualizing the contributions of this research. The chapter begins by surveying core techniques in information extraction, particularly methods for identifying clinical entities and the relationships between them within free-text records. Subsequently, the review turns to two key applications of clinical NLP: medical text classification and drug efficiency detection. These tasks rely heavily on robust modeling strategies to process clinical abstracts and patient-generated reviews,

respectively. Representative studies are analyzed to highlight how predictive models have been trained to classify diseases or assess drug responses based on free-text data. The next section explores text vectorization strategies, additionally, the chapter reviews NLP techniques. Finally, the chapter presents a detailed overview of the role of ontologies in healthcare. By synthesizing these strands of research, the chapter establishes the theoretical and technical foundation for the proposed contributions of this thesis.

2.2 Information Extraction

Information extraction (IE) in healthcare is a rapidly advancing field aimed at automating the extraction of meaningful data from both structured and unstructured medical sources. The vast majority of healthcare data exists in an unstructured format, such as clinical notes, patient records, and research publications, all of which present significant challenges for analysis and decision-making. In contrast, structured data, such as demographic information, lab results, and standardized coding (e.g., ICD-10 codes), is already organized in databases or EHRs, making it more straightforward to process. However, the real challenge lies in extracting valuable insights from the unstructured clinical text that accounts for the majority of information within the healthcare system.

With the advent of ML techniques, the extraction of information from clinical text experienced significant improvements. Unlike rule-based systems, ML approaches are able to learn from data, identifying patterns and making predictions based on annotated examples. Conditional Random Fields (CRFs) and Support Vector Machines (SVMs) became commonly used for named entity recognition (NER) and other IE tasks, enabling automated extraction of clinical entities, such as diagnoses, medications, and procedures, from EHRs and clinical notes [67]. These models typically rely on manually annotated training data and can handle a variety of linguistic structures. However, they still faced limitations in terms of feature engineering and required large amounts of labeled data to perform well across diverse text types. Recent advancements in DL have revolutionized information extraction from unstructured clinical text. Deep learning models, particularly those based on Recurrent Neural Networks (RNNs) and Transformers, have demonstrated superior performance in understanding and processing complex medical narratives. Models such as BERT have shown remarkable success in handling unstructured text by capturing context-dependent relationships within the data. In healthcare, ClinicalBERT, a version of BERT fine-tuned on clinical corpora, has been applied to tasks such as entity recognition and classification of clinical concepts, including diseases and treatments [68]. These models have the ability to understand both the context and relationships between entities, significantly improving

the accuracy of clinical data extraction. Additionally, DL techniques have enhanced relation extraction, where the objective is to identify and categorize relationships between extracted entities. For instance, identifying whether a medication is prescribed for a particular medical condition or if a patient has been referred to a specialist. These advancements have allowed models to extract more nuanced information from clinical text, improving clinical decision support systems [69].

Information extraction (IE) from EHRs is a fundamental area in biomedical natural language processing, aiming to transform unstructured clinical narratives into structured, actionable data. Text mining techniques have been effectively applied to detect critical clinical information, such as adverse drug events (ADEs) [70]. Advances in sequence labeling have also contributed to improving information extraction, including the integration of global label context within recurrent neural networks [71].

In summary, clinical entity extraction is a critical task in the medical field, involving the identification and categorization of relevant medical entities from clinical texts. This process is essential for improving information retrieval, decision-making, and the development of clinical practice guidelines. Various methodologies have been proposed to enhance the accuracy and efficiency of clinical entity extraction, leveraging advanced ML and NLP techniques.

2.3 Events, Temporal Expression, and TLink

CNNs have proven to be highly effective for classifying medical texts, leveraging deep architectures to facilitate the automatic extraction of relevant information from clinical narratives. This method significantly alleviates the reliance on manual annotation and handcrafted features. For instance, in a prior study [4], CNNs were applied within a weakly supervised clinical classification framework utilizing pre-trained word embeddings and rule-based label generation. The model was evaluated alongside traditional classifiers such as Support Vector Machines (SVM), Random Forests, and Multilayer Perceptrons, with CNNs yielding the highest F1-scores (0.92 for smoking status and 0.97 for fracture detection) across datasets from the Mayo Clinic and i2b2. Comparative evaluations have further demonstrated that CNNs can outperform traditional NLP techniques by approximately 15% in sentence-level classification tasks [72]. Moreover, integrating rule-based elements such as trigger terms and Unified Medical Language System (UMLS) embeddings with CNNs has been shown to improve clinical text classification accuracy [73]. In the context of more recent models, [74] explored a BERT-based encoder coupled with a feed-forward network, achieving an F1-score of 83.5% on the i2b2-2012 dataset. Other efforts, such as [75], utilized rule-based approaches

and Conditional Random Fields (CRF) to identify clinical events, while Random Forests were employed to determine event attributes like polarity and modality. Tools like HeidelTime played a central role in identifying temporal expressions. Hybrid systems combining rule-based and ML techniques have also been effective. For instance, [76] combined Naive Bayes models (via Weka) with HeidelTime and semi-supervised CRFs to extract temporal expressions and events, leveraging the GENIA tagger for lemmatization and syntactic processing. Similarly, [77] employed temporal CNNs and additional NLP components like the RegexpTokenizer and PerceptronTagger to detect event spans. In SemEval 2017, [78] proposed a hybrid model combining rule-based methods with SVMs and RNNs to extract temporal information. The MedTime system [79] also incorporated both rule-based and ML approaches, including CRFs and SVMs, in conjunction with HeidelTime for robust temporal tagging on the i2b2-2012 dataset. A notable architecture proposed in [80] utilized a Bi-LSTM-CRF network, combining CNNs for subword representation, contextual embeddings, and medical knowledge before applying CRF for final sequence labeling. Efforts such as [81] have investigated diverse text representation techniques (Word2Vec, GloVe, fastText, ELMo, BERT, and BioBERT) for medical concept extraction. However, in datasets like i2b2 2010 and 2012, temporal expressions were not the primary focus, which were instead centered on clinical entities such as "PROBLEM", "TEST", and "TREATMENT" for i2b2 2010. While many studies have addressed event and temporal expression extraction using ML and DL techniques, recent trends emphasize the integration of domain ontologies to improve semantic precision. Nonetheless, challenges persist in fully harnessing these ontologies given the complexity and variability of medical language. Thi thesis addresses this gap by introducing a novel approach that fuses DL techniques with ontological reasoning to enhance the extraction and contextual interpretation of clinical events and temporal expressions in narrative medical texts.

Prior to the widespread adoption of DL, relation classification tasks in clinical narratives primarily relied on traditional statistical ML techniques. For instance, the work in [76] utilized the i2b2-2012 dataset to classify temporal relations—specifically, BEFORE, AFTER, and OVERLAP—between pairs of clinical events using a Naive Bayes classifier implemented via the Weka toolkit. A comprehensive NLP framework for extracting events, temporal expressions, and temporal relations from clinical records was proposed in [82]. This system integrated unsupervised, supervised, and rule-based approaches. Supervised components included CRFs and support vector machines (SVMs), with automated feature selection employed to optimize performance. The unsupervised elements involved Brown clustering, rendering the overall system semi-supervised. Their work focused on a simplified temporal relation taxonomy (AFTER, BEFORE, and OVERLAP) of i2b2-2012 dataset. In another

notable contribution, the study in [83] extended the HeidelTime system to enhance the recognition of temporal expressions within clinical texts. To address the extraction of temporal links (TLinks) from i2b2-2012 data, the authors of [84] proposed a hybrid approach combining rule-based methods and ML. Their system was designed to detect BEFORE, AFTER, and OVERLAP relations. Further advancements were presented in [85], where the authors introduced an end-to-end temporal relation extraction system. This framework employed multiple SVM classifiers in conjunction with Markov Logic Networks (MLNs) to infer temporal relationships across clinical events. The work in [86] approached the problem using probabilistic soft logic to globally infer temporal relations by modeling time graphs. Text representations were obtained via a BERT encoder; however, their focus remained on relation extraction rather than entity recognition.

A DL-based method using an Attention-based Bidirectional Long Short-Term Memory network (Att-BiLSTM) was proposed in [87]. This model integrated both domain-specific word embeddings derived from MIMIC and GloVe embeddings, and was trained to identify the temporal relation classes—AFTER, BEFORE, and OVERLAP. Lastly, the framework introduced in [88] combined NER and relation extraction modules. Built on top of Spark NLP and its pre-trained NER models [89], the system incorporated two distinct relation extraction architectures: one leveraging BioBERT and the other based on a Fully Connected Neural Network (FCNN), facilitating effective end-to-end temporal relation extraction. Table 2.1 presents the main studies that have utilized the i2b2-2012 dataset.

Table 2.1: Summary of the key studies and their respective models that achieved the best performance on the i2b2-2012 dataset.

Information Extracted	**Approach**	**Study**
Event	BERTLARGE + BiLSTM-CRF	[81] (2019)
	Bi-LSTM-CRF	[80] (2022)
Event + TIMEX	Bio+ Clinical BERT + UMLS	[90] (2021)
	UMLS + BERT	[91] (2021)
	Large Language Models (LLMs)	[92] (2025)
Relation_3	BERT + Feed-Forward Network + Probabilistic Soft Logic(PSL	[86] (2021)
	Att-BiLSTM, word embeddings using GloVe mbeddings	[87] (2021)
Relation_8	BioBERT	[88] (2021)
Event + Relation_3	BERT-based + feed-forward network	[74] (2023)
Event + TIMEX + Relation_3	CRFs + SVM (Event); HeidelTime + CRFs + SVM (TIMEX); Rule-based + SVM (Relation_3)	[93] (2013)
	Rule-based + CRF (Event); Rule-based (TIMEX); Rule-based + ML (Relation_3)	[84] (2013)
	REBEL (BART-based)	[94] 2023
Event + TIMEX + Relation_8	Rules + CRF (Event); HeidelTime (TIMEX); LR, NB, and DT (Relation_8)	[75] (2013)
	HeidelTime + CRF semi-supervised (Event+TIMEX); Naive Bayes + CRFs + SVM (Relation_8)	[76] (2019)

2.4 Detection of Drugs Efficiency

Patient reviews are an invaluable source of insight for understanding experiences and outcomes related to medical treatments and medications. However, these reviews are often characterized by informal language, specific medical terminology, and a high degree of lexical diversity, presenting unique challenges for analysis. Informal language, frequently used in patient reviews, can include colloquial expressions, abbreviations, and non-standard grammar, making it difficult for traditional analytical models to extract meaningful patterns or classify data accurately. Additionally, medical terminology used by patients may not align perfectly with clinical standards, further complicating the interpretation process. The high lexical diversity in patient reviews arises from the variety of ways individuals describe similar experiences. For instance, side effects of a medication may be described using different phrases or expressions depending on the patient's background, health literacy, or personal experience. This diversity necessitates robust text preprocessing techniques to normalize language and reduce noise in the data. Furthermore, patient reviews often blend subjective sentiments with objective details, requiring models to distinguish between emotional opinions and factual observations to provide reliable analyses. Addressing these challenges requires the implementation of advanced NLP methods, such as word embeddings, which can capture the contextual meaning of terms, even in informal or diverse linguistic environments. By acknowledging and addressing the complexities inherent in patient reviews, researchers can develop more effective models to analyze these texts, ultimately contributing to improved patient care, drug development, and public health decision-making.

To address the challenges presented above, some researchers have proposed advanced methods for processing such data. For instance, the authors in [95] introduced a text vectorization method using TF-IDF combined with FastText word embeddings, alongside five classifiers: SVM, DT, RF, Naïve Bayes, and KNN. Colón-Ruiz et al. [96] advanced this research with a system based on word embedding and LSTM networks, achieving an impressive accuracy of 0.9046 compared to alternative models such as CNN and hybrid CNN-LSTM architectures. Gräßer et al. [97] evaluated model performance using data filtered for particular conditions, selecting five common disorders from the Druglib.com dataset—Depression, Pain, Anxiety, Contraception, and Type 2 Diabetes. Their logistic regression model on Drugs.com achieved a remarkable accuracy of 0.9224. In another approach, a study utilized TF-IDF principles with a Roberta classifier enhanced by Data Augmentation and Inference Augmentation (DAIA), though the achieved accuracy was limited to 0.316 [98]. Additionally, Bemila et al. [99] applied the BOW method with algorithms such as Naïve Bayes, Random Forest, Linear Support Vector Classification (SVC), Logistic Regression (LR), and

RNN-BiLSTM (Recurrent Neural Networks with Bidirectional Long Short-Term Memory). Innovative models have also emerged to address these challenges. For example, MIN, Z. [100] employed the AskaPatient dataset to develop a Weakly Supervised Mechanism combining CNN and LSTM (WSM-CNN-LSTM). This hybrid model achieved an accuracy of 86.72%, underscoring the potential of advanced DL approaches to analyze patient reviews effectively. These studies collectively highlight the importance of leveraging both linguistic and medical insights to understand patient experiences, emphasizing the growing role of ML and NLP in the healthcare domain.

2.5 Clinical Documents Classification

Medical text classification continues to attract significant attention within the NLP research community, particularly due to its potential in biomedical applications. Numerous approaches have been introduced in recent years to address classification tasks using benchmark datasets such as Ohsumed. For example, the study in [101] employed a Convolutional Neural Network (CNN) architecture for multi-label classification on the Ohsumed-20000 dataset, reporting a precision score of 36.0. In the same work, a second architecture combining CNN and LSTM layers achieved an improved accuracy of 37.5. These models utilized word representations derived from Word2Vec embeddings trained on the Google News corpus. Another investigation [102] restricted the classification task to the top 10 categories out of the 23 available in the Ohsumed dataset. In [103], researchers introduced a Knowledge-enhanced Spherical Representation Learning approach, attaining an accuracy of 69.24% by employing Joint Spherical Embedding (JoSE) [104]. Furthermore, in [105], a subset of 7,400 Ohsumed documents was used to develop the BertGCN model, which integrates BERT for document vectorization with Graph Convolutional Networks (GCN) for classification. An alternative approach is presented in [106], where the authors proposed a novel Graph Neural Network (GNN)-based method. Rather than constructing a global corpus-level graph, their strategy involves creating a localized text-level graph for each document by linking words within a predefined context window. Initialized with GloVe embeddings, their model achieved a precision of 69.4 ± 0.6. When comparing traditional methods, TF-IDF-based logistic regression outperformed CNNs with randomly initialized embeddings, especially on longer documents such as those in the 20 Newsgroups dataset (20NG). In a comparative analysis [107], the authors explored the impact of three text representation techniques on classification performance using the Ohsumed corpus: TF-IDF [108], Latent Dirichlet Allocation (LDA) [109], and Word2Vec. Their results demonstrated that representation choice significantly influences classifier performance. Lastly, Sowmya et al. [110] conducted a study

on multi-label classification, applying TF-IDF encoding with both the Rocchio algorithm and kNN for effective document categorization.

2.6 Text Vectorization

Transforming textual data into a digital format is a fundamental requirement for applying data mining models. One of the earliest methods for this transformation is the One-Hot encoding representation, where each word is represented by a unique binary vector. For a vocabulary of size V, each word w_i is encoded as a vector with 1 at the i^{th} position and 0 elsewhere, independent of the word's context [111]. In [112], the authors explored region embedding utilizing one-hot representation. By concatenating one-hot vectors, the Bag of Words (BOW) model is formed. However, its primary limitation lies in the significant length of the resulting characteristic vector [113].

An enhancement to this method is Term Frequency-Inverse Document Frequency (TF-IDF), which builds on both Term Frequency and Inverse Document Frequency [114]. Studies by [114] and [115] suggest that word embedding techniques often outperform traditional methods like topic modeling and TF-IDF.

Word embeddings have gained prominence for tasks such as machine translation, chatbots, image captioning, and language modeling. These techniques offer semantic representations, unlike earlier methods. Word vector models can be categorized into global matrix factorization-based models and local context window-based models. Examples include Latent Semantic Analysis (LSA) [116], Skip-Gram, and Continuous Bag of Words (CBOW) models. LSA maps words to concepts by analyzing word-document relationships, while Word2Vec groups words with similar contexts and separates unrelated ones. The authors in [117] applied Word2Vec to various datasets, achieving a high accuracy of 0.9342 for COVID-19 datasets and a low of 0.3164 for Drugs.com datasets. Unlike One-Hot, BOW, or co-occurrence matrices, Word2Vec vectors are independent of vocabulary size [111]. The CBOW approach predicts a word based on its context and has been applied to gene and protein synonym recognition [118] and word embedding pretraining using datasets MIMIC-III [119]. To address polysemy, pre-trained embeddings like CBOW and Skip-Gram are used to create distinct word vectors for different meanings [120].

The Doc2Vec model [121] extends Word2Vec by providing representations for sentences, paragraphs, and entire documents, while GloVe [122] uses global matrix factorization to leverage corpus-wide word co-occurrence statistics. GloVe has been used to assign ICD-9 codes from clinical records [123]. Region embedding, or two-view embedding, combines word and context perspectives [112].

Recent vectorization methods include transformer-based architectures like BERT [124], which uses attention mechanisms to capture contextual word relationships. Its encoder-decoder structure provides token representations conditioned on left and right contexts. Variants like SciBERT [125], BlueBERT [126], and BioBERT [127] are tailored to scientific, medical, and biological domains, respectively. ELMo [128] employs bidirectional LSTMs for dynamic and context-aware word embeddings.

Meta-embedding learning combines multiple embedding sets, offering improved performance and vocabulary coverage [129]. Topic embeddings, as used in [114], enable document representation through topic sets.

2.7 Natural Language Preprocessing

The characteristics of textual data highlight the essential role of preprocessing in improving data quality and optimizing the effectiveness of data mining processes. Different preprocessing approaches are applied depending on the data type. For structured data, preprocessing typically includes cleaning, integration, transformation, and reduction. Conversely, semi-structured and unstructured data require specific techniques such as tokenization, stop-word removal, stemming, and vectorization methods.

In the medical field, the sensitivity and complexity of preprocessing are significantly heightened due to challenges like medical codes, abbreviations, spelling errors, and negations. For instance, medical record spelling errors are reported at a high rate of approximately 10% [130]. According to [131], preprocessing medical texts involves standardizing terms and acronyms using regular expressions and procedural methods before applying vectorization techniques to EHR. as well as several studies have reviewed data processing and text mining technologies on EMRs, emphasizing the need for robust preprocessing and domain-specific adaptation [132]

Information extraction tasks, such as NER and relationship extraction, are critical for feature detection. Tools like *Specialist* [133], *LEXTOOL*, and UMLS-based spelling correction systems [134] incorporate linguistic and morphological analysis along with spelling correction. Negation detection tools, such as *NegExpander* [135], play an essential role in medical text analysis, while temporal analysis, as proposed in [136], often relies on complex temporal models. Modular architectures integrating NLP techniques with knowledge bases have also been proposed for processing temporal information in clinical narrative documents [137, 135].

The text further highlights the importance of data augmentation for small datasets and advocates for dimensionality reduction techniques. ML approaches, including Principal Com-

ponent Analysis (PCA), Latent Semantic Analysis (LSA), and Latent Dirichlet Allocation (LDA), are recommended for this purpose [120].

2.8 Ontologies in healthcare

The adoption of ML technologies in digital health applications offers immense potential; however, the requirement for extensive, hand-annotated datasets presents a major challenge to their broader implementation [138]. Automating the disambiguation of clinical documents, such as resolving abbreviations, requires precise annotations, which exacerbates the imbalance of labeled training data and hinders the deployment of ML across clinical and research workflows. Effective knowledge management in digital health, crucial for patient well-being, faces difficulties due to semantic misinterpretations that can result in medical errors, adversely affecting health outcomes. As highlighted by researchers [139, 140, 141, 142, 143], ontologies are essential for recognizing entities, detecting implicit knowledge, and extracting semantic insights from clinical narratives. Unlike thesauri or taxonomies, ontologies provide a structured framework for conceptual meanings, enabling advanced reasoning and knowledge generation. By defining shared conceptualizations, ontologies allow for sophisticated mappings between domains such as anatomy, diseases, phenotypes, and laboratory investigations [144]. Furthermore, ontologies such as the Ontology for Biomedical Investigations (OBI) have played a crucial role in supporting semantic enrichment and interoperability of extracted biomedical knowledge [145].

The ambiguities inherent in word meanings, such as the term ”mole,” underline the importance of context-aware ontology mappings [146, 147]. Medical ontologies like SNOMED-CT, Human Phenotype Ontology, UMLS, and MeSH significantly enhance structured medical terminologies, improving vector-space representations and relation extraction models [146]. Recent studies demonstrate the benefits of combining ML with ontological methods, particularly in addressing abbreviation disambiguation problems. For instance, utilizing the Finley database and CASI dataset resulted in a 3% improvement in outcomes [146]. Ontologies also contribute to structured knowledge representation, as seen in projects like GALEN, which facilitates the consolidation of simulation databases [139]. Beyond healthcare, ontologies find applications in diverse fields, including dependency extraction with active learning [148], developing pedagogical models in intelligent tutoring systems [149], and identifying gene/protein synonyms for social media analysis [123]. Biomedical ontologies, such as the Cardiovascular Disease Ontology (CVDO), enhance word embeddings and support advanced NER architectures in healthcare [123] and [118]. Leveraging UMLS-based ontologies further facilitates semantic annotation of biomedical texts [150] and supports semantic relation ex-

traction in studies such as those focusing on breast cancer treatment [142]. Ontologies thus provide well-defined models of medical concepts and relationships, improving data interpretation and integration. The PSI framework [151], using a medical knowledge graph, and DiseaseNet's integration of structured and unstructured data exemplify the application of ontological principles to represent and relate medical entities effectively.

Word2Vec embeddings, when enhanced with ontology-based information, further demonstrate the value of integrating structured knowledge into ML models. For example, Word2Vec embeddings enriched with UMLS information showed a Spearman's correlation of 0.733 with the mini-Mayo dataset and improved performance in NER tasks [152]. While Bio + Clinical BERT achieved superior F1-scores of 0.87 and 0.811 on the i2b2 2010 and 2012 datasets, respectively, Word2Vec embeddings remained effective for capturing lexical and clinical relationships. Additionally, UmlsBERT, a contextual embedding model, incorporates structured biomedical knowledge during pre-training. Unlike BioBERT and Bio_ClinicalBERT, which focus solely on domain-specific corpora, UmlsBERT integrates expert knowledge from UMLS by linking words with the same underlying concepts and employing semantic type information. This methodology enabled UmlsBERT to achieve top F1-scores on the i2b2-2006, 2010, and 2012 tasks (93.6%, 88.6%, and 79.4%, respectively) and the best accuracy on the MedNLI task (83.0%) [91]. Although it did not outperform BERT on the i2b2-2014 task, UmlsBERT surpassed other biomedical BERT models, highlighting the benefits of integrating domain-specific knowledge into contextual embeddings. Ablation tests further confirmed the positive impact of semantic type embeddings on model effectiveness across various datasets. In conclusion, the integration of ontologies in healthcare and biomedical domains plays a pivotal role in overcoming challenges related to semantics, knowledge representation, and interoperability. When combined with ML, ontologies drive advancements in automated processing, decision support, and information extraction from clinical narratives. As demonstrated by recent studies, the interplay between ML and ontology-based methodologies continues to propel innovation in the digital health landscape.

2.9 Conclusion

This chapter has presented a critical review of the foundational and contemporary literature that underpins the research directions pursued in this thesis. By examining key contributions in clinical information extraction, medical text classification, and drug efficiency detection, the chapter highlights the methodological advancements and persistent challenges in applying NLP to healthcare data. The discussion of text vectorization and preprocessing techniques emphasizes their essential role in transforming unstructured clinical narratives

into formats amenable to ML and DL models. Additionally, the exploration of ontologies in the healthcare domain demonstrates their importance in enabling semantic interoperability, standardizing medical terminology, and supporting context-aware reasoning. Collectively, these insights establish a solid foundation for the proposed approaches developed in subsequent chapters. They also underscore the necessity of integrating linguistic, statistical, and semantic resources to enhance the performance, interpretability, and applicability of clinical NLP systems.

Chapter 3

New approachs for Events, TIMEX and Tlink extraction

Contents

3.1 Introduction 51

3.2 New DL Model for Events and TIMEX extraction 51

3.2.1 Data Preprocessing 53

3.2.2 BERT Embeddings 56

3.2.3 Models Description 57

3.2.4 Evaluation Metrics 61

3.2.5 Results 62

3.2.6 Prediction Process 69

3.3 An advanced approach using Ontologies and DL for Events and TIMEX extraction 70

3.3.1 Ontologies Description 70

3.3.2 Key Biomedical and Clinical Entity Types 72

3.3.3 Ontologies using Jena API and Deep Learning 73

3.3.4 Ontologies using Owlready2 and DL 91

3.3.5 Jena API vs. Owlready2 103

3.4 New Approach for temporal Relations Extraction 104

3.4.1 Dataset and Pre-Processing Process 105

3.4.2 Models Description 106

3.4.3 Results and discussion 108

3.4.4 Temporal Link (TLink) Prediction Approach 113
3.5 Conclusion 118

3.1 Introduction

This chapter presents a comprehensive investigation into the extraction of clinical events, temporal expressions (TIMEX), and TLink from unstructured medical narratives, with a particular emphasis on DL techniques and ontology-based semantic reasoning. This chapter is structured into three main components, each progressively contributing to the enhancement of clinical information extraction in narrative texts. The first part focuses on the use of DL models for event and temporal expression extraction. The second part explores the integration of DL with semantic technologies to further enhance extraction accuracy and contextual understanding. Two ontology frameworks are studied. First, the Jena API is employed for managing OWL ontologies and executing SPARQL queries to enrich semantic reasoning in clinical text. This integration allows for the logical inference of entity types and supports ontology-driven annotation. Second, the Owlready2 library is introduced as a Python-based alternative for reasoning and ontology querying. Both tools are evaluated in conjunction with the CONSEPTR DL model across multiple clinical datasets. Comparative insights between the two frameworks highlight their strengths in enabling knowledge-driven NLP applications.

The final part addresses temporal relation extraction, where the proposed model combines outputs from standard, separable, and transposed convolutional networks through majority voting to classify temporal links between entities. This approach extends beyond entity detection to infer relational structure, offering deeper temporal understanding critical to clinical reasoning. In summary, this chapter underscores the synergistic potential of combining DL with ontology-based reasoning for medical temporal information extraction. The methodologies proposed demonstrate robust generalization across datasets and offer scalable solutions for enhancing semantic interpretation in clinical NLP.

3.2 New DL Model for Events and TIMEX extraction

The extraction of events and temporal expressions is formulated as a classification problem. The system identifies entities belonging to the predefined categories of Event and TIMEX. Additionally, a special type, TOKEN, is introduced to annotate entities that do not belong to either category. Events encompass a range of clinical elements, including medical entities

such as problems, tests, and treatments; clinical department references indicating the units or services associated with a patient's admission status; evidential indicators—phrases that reflect information sources (e.g., "complained" in "The patient complained of a headache lasting one week"); and occurrences, which refer to administrative or procedural actions like admission, transfer, or discharge that impact the patient's clinical timeline.

To extract entity types the functions, G is defined as follows: $G(x_i) = y \quad$ where $y \in E, \quad x_i \in X, \quad X = E \cup \{TOKEN\}$,
and $E = \{Event \cup TIMEX\}$, Knowing:
Event = {CLINICAL_DEPT, EVIDENTIAL, OCCURRENCE, TEST, TREATMENT, PROBLEM}.
TIMEX = {TIME, FREQUENCY, DATE, DURATION}.

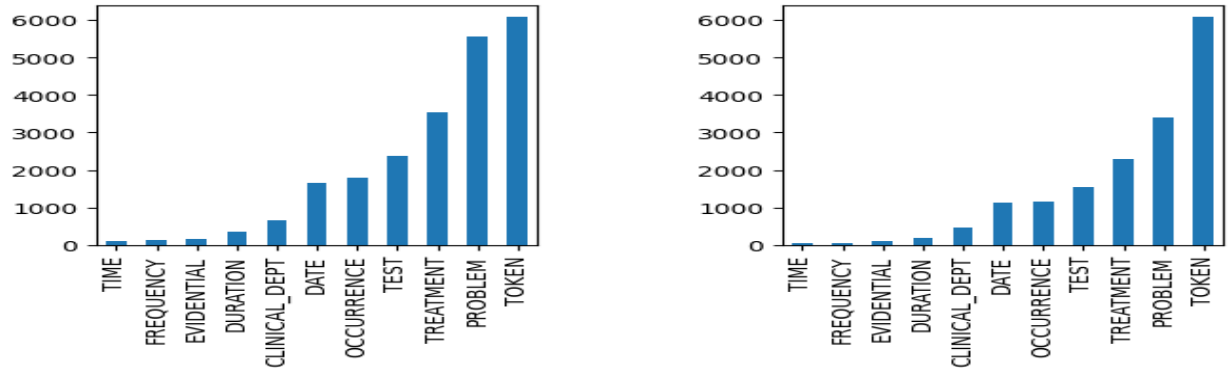
Data Description: Model training was conducted using the i2b2-2012 dataset, while evaluation was carried out across three benchmark datasets: i2b2-2012, i2b2-2010, and i2b2-2018, in order to assess generalizability across temporal corpora. All datasets used are publicly available upon request. Access requires registration through the DBMI portal[1], after which researchers may submit dataset requests via the n2c2 NLP research data sets page[2].

```
Admission Date :
08/08/2016
Discharge Date :
15/08/2016
HISTORY OF PRESENT ILLNESS :
The patient is a 47 year old woman with type 1 diabetes mellitus ,
who took postcadaveric kidney transplantation  for four months
and now has good graft function . She presents for cadaveric
pancreas transplantation . Her diabetes mellitus has been
complicated by retinopathy and nephropathy in common with
peripheral neuropathy .
```

The presented example (narrative medical text) demonstrates the annotation of events and temporal expressions within the i2b2-2012 dataset. In this instance, the terms Admission and Discharge are categorized as *OCCURRENCE* events, while retinopathy and nephropathy are classified as *PROBLEM* entities. Additionally, postcadaveric is labeled as a *TREATMENT* entity. TIMEX are annotated according to their type, which can be classified as one of the following: *TIME*, *DATE*, *FREQUENCY*, or *DURATION*. In the provided example, the temporal expressions and their corresponding types are as follows: (four months, DURATION), (08/08/2016, DATE), (15/08/2016, DATE).

[1]https://portal.dbmi.hms.harvard.edu/
[2]https://portal.dbmi.hms.harvard.edu/projects/n2c2-nlp/

i2b2-2012 Corpus: The i2b2-2012 corpus consists of a total of 310 discharge summaries, with 190 records allocated for training and 120 for testing. The annotations in this dataset were manually performed by eight annotators, four of whom had a medical background, as described in [153]. The annotation process was conducted using the Multi-purpose Annotation Environment (MAE) and the Multi-document Adjudication Interface (MAI) toolkits [154]. Figure 3.1 presents the train (Figure 3.1a) and test (Figure 3.1b) datasets after preprocessing process.



(a) Tain Datasets. (b) Test Dataset.

Figure 3.1: Train and Test datasets for i2b2-2012 coprpus after preprocessing.

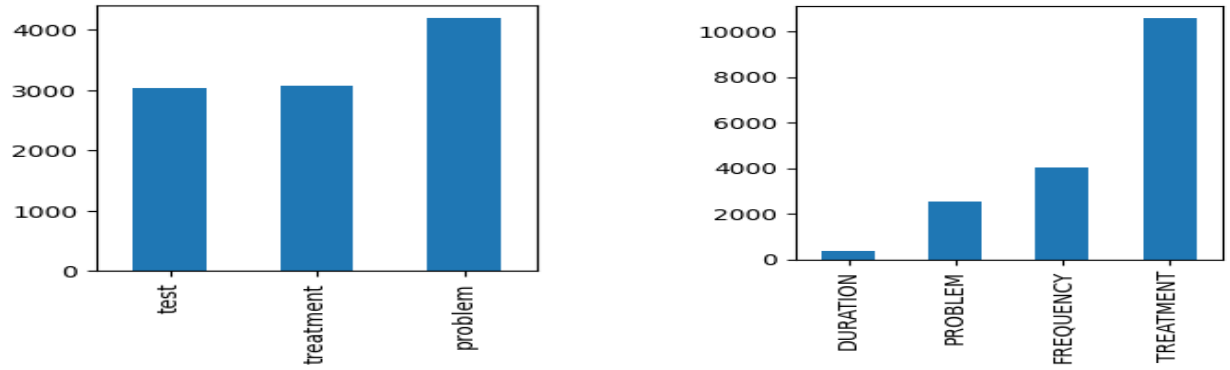
i2b2-2010 Corpus: The i2b2-2010 dataset consists of 256 clinical reports, specifically designed to evaluate the model's ability to extract medical concept types, as provided by the challenge [155].

i2b2-2018 Corpus: The test dataset from Track 2 of the i2b2-2018 challenge has been used, which focuses on adverse drug events and medication extraction. The dataset includes a total of 202 clinical reports [156].

The i2b2-2012 corpus, which serves as the foundation for both the training and testing datasets. Figure 3.2 presents the i2b2-2010 test dataset (Figure 3.2a) and i2b2-2018 test dataset (Figure 3.2b) datasets after preprocessing process.

3.2.1 Data Preprocessing

The following examples illustrate how numerical information in clinical narratives can lead to different disease subclassifications. When the dosage numbers are removed, the texts are



(a) i2b2-2010 Test Dataset.

(b) i2b2-2018 Test Dataset.

Figure 3.2: Test datasets for i2b2-2010 and i2b2-2018 coprpora after preprocessing.

more uniformly classified under a general disease category.

1. **Stage 1 (Mild) Hypertension:** A 62-year-old male presents with average blood pressure of 145/92 mmHg and reports occasional dizziness. The treatment includes Lisinopril 10 mg and Hydrochlorothiazide 12.5 mg, administered once daily.

2. **Stage 2 (Intermediate) Hypertension:** A 62-year-old male presents with persistent blood pressure around 155/98 mmHg, accompanied by dizziness. Treatment is adjusted to Lisinopril 10 mg and Hydrochlorothiazide 25 mg.

3. **Stage 2 (Severe) Hypertension:** A 62-year-old male presents with uncontrolled blood pressure despite initial therapy, with measurements reaching 165/105 mmHg, treatment is intensified to Lisinopril 20 mg and Hydrochlorothiazide 25 mg once daily.

These examples illustrate a progression of hypertension severity in a 62-year-old male patient, with each entry detailing specific blood pressure readings and corresponding medication dosages. These numerical values are pivotal in distinguishing between different stages of hypertension and tailoring appropriate treatment plans. Removing numerical data from such clinical narratives can significantly impair the accuracy of medical text classification and diagnosis. Numerical information, such as blood pressure readings and medication dosages, provides essential context that aids in the precise categorization of medical conditions. For instance, distinguishing between Stage 1 and Stage 2 hypertension relies heavily on specific blood pressure thresholds.

In summary, preserving dosage information in medical narratives allows the extraction process to capture subtle differences in treatment intensity, which can lead to differentiated classifications of Hypertension. This observation underscores how critical preprocessing decisions (whether it involves deleting numbers, applying stemming or lemmatization, or

removing extraneous symbols) can substantially impact the granularity and accuracy of text classification outcomes.

In contrast to other approaches, during text preprocessing, all the following steps are avoided, as they may damage the syntactic or semantic information: Removing punctuation marks, dots, commas, question marks, and numerical values like ”165/105” and ”25 mg” from this corpus can alter the significance of a patient's medical history. Additionally, applying lexical processing techniques such as lemmatization or stemming, which involve reducing words to their base or root forms, can undeniably lead to changes in temporal knowledge. Converting all past tenses to their corresponding infinitive forms can alter the meaning of a patient's medical history and the surrounding context [157]. The preprocessing process can be summarized as follows:

For i2b2-2012 dataset:

- The original train and test datasets are merged.
- Entities are split using balanced sampling into training (75%) and testing (25%).
- A *TOKEN* entity is added to both the training and testing sets.
- Duplicate entities are removed to ensure data integrity.

Figure 3.3 provides an overview of the datasets used in this study and the preprocessing steps applied.

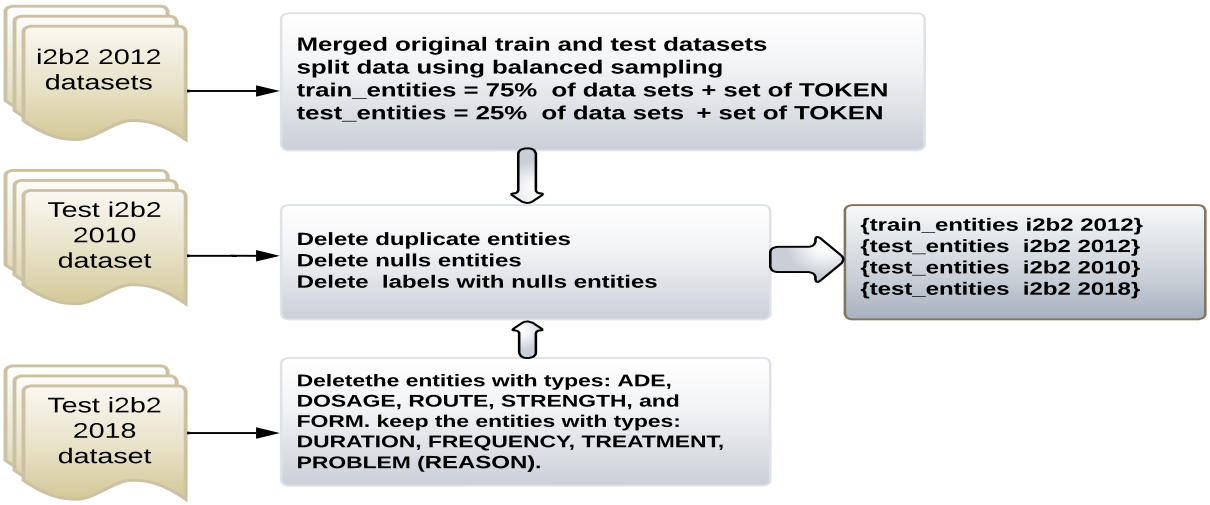


Figure 3.3: Description of data preprocessing.

For i2b2-2010 and i2b2-2018 datasets: A slightly different approach is adopted. Instead of merging train and test datasets, ‘test_entities‘ are directly extracted from their

respective test datasets. As with the i2b2-2012 dataset, duplicate entities are systematically eliminated to uphold data quality and consistency.

A specific refinement is applied to the i2b2-2018 dataset, where certain entity categories are excluded from analysis. Specifically, entities labeled as ADE, DOSAGE, ROUTE, STRENGTH, and FORM are omitted. Meanwhile, entities belonging to the categories DURATION, FREQUENCY, TREATMENT, and PROBLEM (REASON) are retained to align with the dataset's unique characteristics and research objectives.

the proposed model can therefore identify the entities defined in the set of Event and TIMEX. Adding a type "TOKEN" allows annotating the type of this entity that does not belong to Event and TIMEX.

3.2.2 BERT Embeddings

BERT is a DL model designed to understand the contextual meaning of words in a sentence by analyzing both left and right contexts simultaneously. Unlike traditional language models that process text sequentially, BERT uses the transformer architecture to capture deep bidirectional relationships. Pre-trained on massive text corpora, it has significantly improved performance in various NLP tasks, including NER, text classification, and question answering. By leveraging transfer learning, BERT can be fine-tuned on domain-specific datasets, making it highly adaptable for different applications. To generate numerical representations of the text, two variants of the BERT model were examined: BERT_en_uncased_L-2_H-128_A-2/1 and BERT_en_uncased_L-2_H-768_A-12/1, as described by Turc et al. [158].

BERT_en_uncased_L-2_H-128_A-2/1: For simplicity, we refer to BERT_en_uncased_L-2_H-128_A-2/1 as BERT_128 throughout this thesis. BERT_128 is a smaller and more computationally efficient variant of BERT[3]. The naming convention indicates that this model has 2 layers (L=2: Number of Transformer layers (encoder blocks)), 128 hidden units per layer (H=128: Hidden size (dimensionality of embeddings and hidden representations)), and 2 self-attention heads (A=2). Compared to the standard BERT model, which has significantly more layers and hidden dimensions, this version is optimized for low-resource environments. While it may not capture as much linguistic depth as its larger counterparts, it is still capable of extracting meaningful representations from text and is particularly useful for tasks requiring quick inference with limited computational power.

[3]https://www.toolify.ai/ai-model/google-bert-uncased-l-2-h-128-a-2

BERT_en_uncased_L-2_H-768_A-12/1: The model BERT_en_uncased_L-2_H-768_A-12/1 is denoted as BERT_768. BERT_768 offers a more robust representation while remaining relatively lightweight. This model consists of 2 layers (L=2), 768 hidden units per layer (H=768), and 12 attention heads (A=12). The increased number of hidden units and attention heads allows it to capture richer contextual information compared to the smaller H-128, A-2 variant. Though it remains significantly smaller than the original $\text{BERT}_{\text{base}}$ (L-12, H-768, A-12), this version strikes a balance between efficiency and performance, making it suitable for tasks that require deeper contextual understanding while still being computationally manageable.

3.2.3 Models Description

For extracting temporal TIMEX and the EVENTs, we transform inputs to numeric values and sort them into several Tensors (i.e., one dimension vector). Thus, the approach relies on a bert-based-uncased preprocessing model (using bert_en_uncased_preprocess/3) and BERT encoder (using BERT_128 or BERT_768)each one into a hub.KerasLayer. To ensure a better information extraction, we employ 128 filters, and the convolution windows equal to 3 with the 'relu' activation function (Figure 3.4). The output of CNN layer uses the flatten layer that makes data in a one dimensional vector. The dense layer in this approach consists of 11 neurons using a sigmoid activation function. Since we handle a multi-class classification, we use a 'categorical_crossentropy', 'adam' optimizer, considering that strides=1, bias=0, and with no padding. Four models have been studied (Standard convolution 1D, Transposed convolution 1D, Separable convolution 1D, and ConvLSTM 1D). But as evaluated and discussed later in this thesis, the extraction of Events and temporal Events rely on the three proposed models based on standard convolution 1D, convolution Transposed and Separable 1D. Algorithm 1 provides an overview of the methodology used for extracting events and temporal expressions.

Standard convolution layer: One of the key advantages of convolution layers is their ability to extract local features from input data, making them well-suited for tasks like image recognition and segmentation. They also provide translation invariance, allowing the network to recognize patterns or features regardless of their position in the input. Moreover, parameter sharing in convolution layers reduces the overall number of model parameters, resulting in more computationally efficient networks that are better at generalizing to new data and classification tasks, including sentiment analysis and topic categorization [159]. These properties make convolution layers crucial for building efficient and effective DL models for computer vision applications, as demonstrated in seminal works such as [160], pioneering

research on document recognition and for image classification [161], breakthrough with deep CNNs.

Transposed convolution layer (Deconvolution: Convolutions that are transposed work by swapping the forward and backward passes. A kernel defines a convolution, while its computing determines whether it is a transposed convolution [162]. It's called "transposed" because it resembles a reverse process of a regular convolution. Deconvolution Networks generate high-resolution output feature maps from low-resolution input feature maps, making them well-suited for semantic segmentation [163]. The Deconvolution flexibility and ability to increase spatial resolution make it a valuable tool in many fields: COVID-19 classification [164], Glioma segmentation [165], and segmenting handwritten historical documents by text lines [85], data augmentation of medical image [166].

Separable 1D convolution layer: Separable convolution is a type of convolution operation that is decomposed into two or more consecutive simpler convolution operations [167]. This decomposition reduces the computational complexity of the convolution while still capturing relevant features from the input data. In contrast, a separable 1D convolution involves breaking down the convolution operation into two stages [168].

1. **Depthwise Convolution:** Each input channel is convolved with a separate filter in this step. However, each filter is only responsible for one channel and is applied independently. It allows the capture of local features in each channel.

2. **Pointwise Convolution:** In this step, a 1x1 filter (filter with only one element) is applied to the outputs of the depthwise convolution. This step combines the information from different channels and performs a linear combination of the features. It helps to create new features that capture interactions between channels.

Separable decreases the computational complexity of convolutional CNNs while maintaining high accuracy by decomposing the convolution operation into two stages: depthwise convolution and pointwise convolution. Moreover, separable convolution requires less memory during training and inference than standard convolutions, which is vital for resource-constrained systems such as mobile devices and embedded systems [169]. The separable convolution is used in a wide variety of applications, including image processing, computer vision, and NLP such as text or image classification and sentiment analysis [170, 171, 172, 173, 174].

ConvLSTM 1D layer: ConvLSTM 1D combines the strengths of CNNs and LSTM networks, enabling it to capture both local patterns and long-range temporal dependencies in

medical text data. Moreover, LSTM components can remember and utilize information from previous words, which is particularly important for understanding the context and evolving medical narratives. The authors in [175], relied on ConvLSTM's potential for improving medical systems, segmenting medical images, and disease detection for electroencephalogram (EEG) signals [176]. ConvLSTM's can capture complex dependencies in meteorological data over both space and time [177]. In video processing, ConvLSTM allows action recognition and video prediction [178]. In NLP tasks, ConvLSTM has been applied for text generation and language modeling [179].

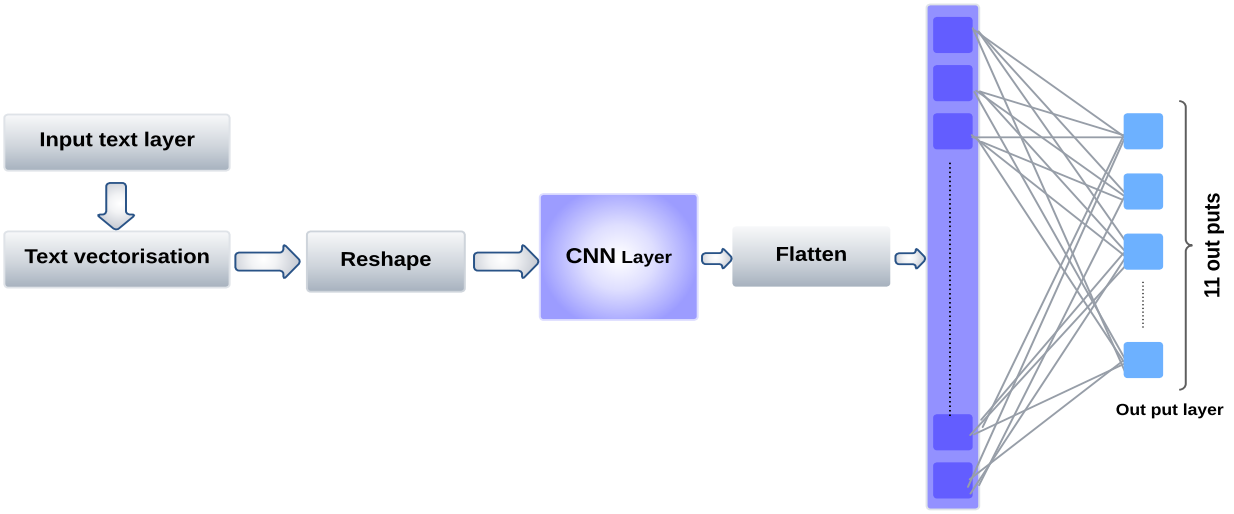


Figure 3.4: The architecture of the basic model, "Text vectorization" is composed of two layers: the first is pretreatment, and the second is coding. "reshape" depends on the type of CNN layer depend on the model used (standard convolution one dimension, separable convolution one dimension, transposed convolution one dimension, or ConvLSTM one dimension). The number of neurons outputs is equal to 11.

Algorithm 1 Train and test the modular models and CONSEPTR model

Require: *train_data*: Training dataset (i2b2_2012)
Require: *test_data*: Test datasets (i2b2_2012, i2b2_2010, i2b2_2018)
Require: *Models*: Set of models {Stand, Trans, Sepr, LSTM}
Ensure: Trained models

▷ Training the models

for each *model* in *Models* **do**
    Train *model* using *train_data*
end for

▷ Testing the models

$t \leftarrow$ Test dataset of *i2b2_2012*
for each *model* in *Models* **do**
    Test *model* using t
end for

▷ CONSEPTR model uses majority voting based on {Stand, Trans, Sepr} ▷ Testing the CONSEPTR model (CONSEPTR is a modular system)

for each t in *test_data* **do**
    Test CONSEPTR using t
end for

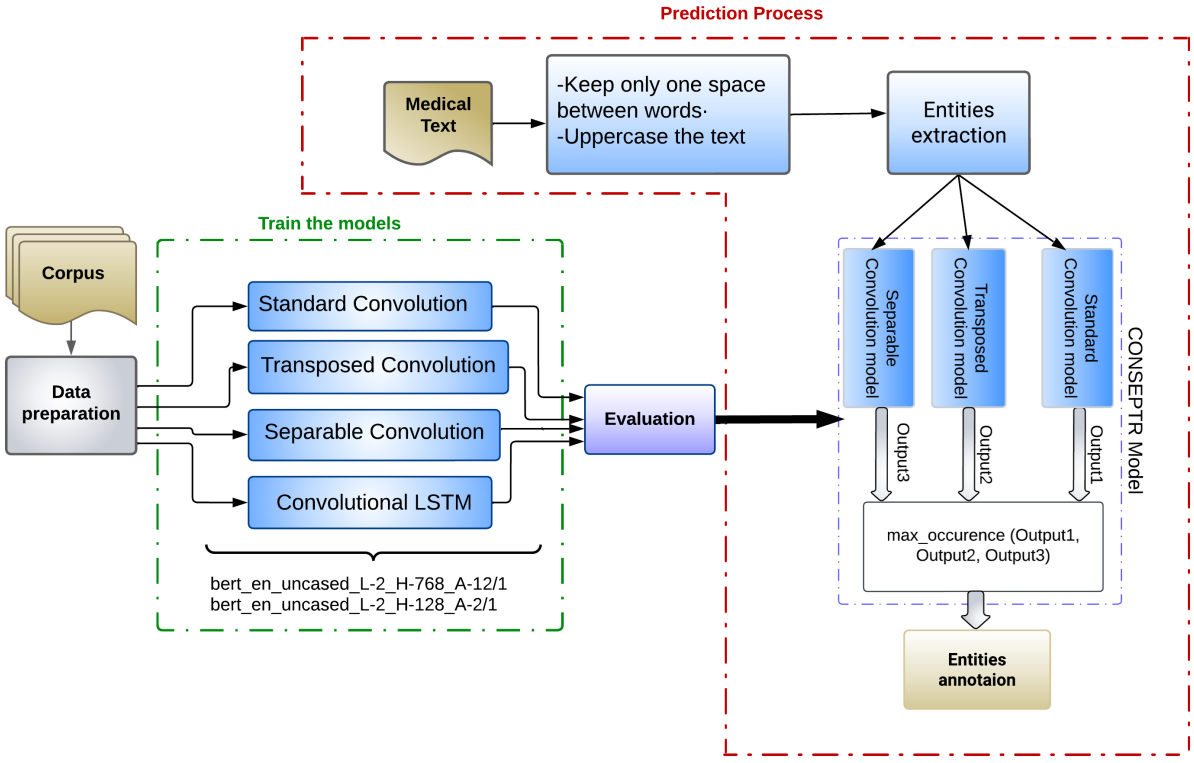


Figure 3.5: A general description of entities extraction methodology, including training and prediction processes.

3.2.4 Evaluation Metrics

To assess the performance of our best model on the test datasets, we employed several key evaluation metrics: precision, recall, accuracy, and the F1-score. Below, we explain each metric in detail.

Precision: Measures the accuracy of the positive predictions made by the model. It answers the question: "When the model predicts a positive result, how often is it actually correct?" High precision means that when a positive prediction is made, it is very likely to be true, thereby minimizing false alarms (false positives).

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.1}$$

Recall: Or sensitivity gauges the model's ability to capture all actual positive cases. In other words, it answers: "Out of all the true positive instances, how many did the model successfully identify?" A high recall indicates that the model misses few true positives, reducing the risk of overlooking important cases.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.2}$$

Accuracy: Represents the overall proportion of correct predictions made by the model, taking both positive and negative cases into account. It answers the question: "How often is the model correct across all classes?" While accuracy provides a general performance measure, it can be misleading if the class distribution is heavily imbalanced.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{3.3}$$

F1-Score: Is the harmonic mean of precision and recall. It balances the trade-off between the two metrics, offering a single measure that reflects both the model's precision and its ability to capture all positive instances. It is particularly useful when you need a comprehensive metric that considers both false positives and false negatives.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.4}$$

Here, TP stands for True Positives, TN for True Negatives, FP for False Positives, and FN for False Negatives.

Based on these evaluations, we identified and saved the model that achieved the highest test accuracy for future predictions.

3.2.5 Results

The results presented in Table 3.1 compare different convolutional architectures (standard (Stand), separable (Sepr), transposed (Trans), and ConvLSTM (LSTM)) on the i2b2-2012 dataset. The performance of these architectures is evaluated using two BERT configurations: BERT_768 and BERT_128. The comparison considers accuracy, precision, and recall across both training and testing datasets.

Overall, the BERT_128 model consistently outperforms the BERT_768 variant. In terms of accuracy, Standard convolution achieves the highest training accuracy (86.90%) in the BERT_128 embedding, followed closely by separable and transposed convolutions. A similar pattern is observed in testing accuracy, where standard convolution (85.27%) outperforms other methods. In contrast, BERT_768 exhibits lower accuracy, with ConvLSTM performing significantly worse compared to other convolution types.

Precision, which reflects the reliability of entity predictions, is relatively lower across all models compared to accuracy and recall. The highest precision is observed in BERT_128, where standard convolution achieves 76.45% precision. On the other hand, ConvLSTM model performs the worst, with precision scores as low as 34.01% in training and 35.39% in testing for BERT_768, indicating an underfitting.

Recall scores demonstrate the models' ability to correctly identify relevant entities. BERT_128 again surpasses the larger variant, achieving near-perfect recall with separable convolution (99.15%) in training and 98.08% in testing. transposed convolution also exhibits high recall, indicating its effectiveness in capturing event and temporal expression structures.

When comparing convolution types, Standard convolution demonstrates the best balance between precision and recall, making it the most effective architecture overall. separable convolution achieves the highest recall but at the cost of lower precision. transposed convolution follows a similar trend to Separable convolution, while ConvLSTM model underperforms across all metrics. Despite its theoretical advantage in capturing sequential dependencies, the ConvLSTM fails to yield competitive precision and accuracy results.

In conclusion, the results highlight the superior performance of BERT_128 in event and temporal expression extraction. While separable and transposed convolutions models improve recall, standard convolution model emerges as the optimal choice due to its balanced performance across all metrics.

Table 3.1: Events and temporal expressions results, the global results for various convolutional models using i2b2-2012 dataset, where Stand: Standrd convolution, Sepr: Separable convolution, Trans: Transposed convolution, and ConvLSTM.

Data	Metric	BERT_768				BERT_128			
		Stand	Sepr	Trans	LSTM	Stand	Sepr	Trans	LSTM
Train	Accuracy	83.85	83.70	83.93	66.27	**86.90**	**86.74**	**86.50**	83.56
	Precision	81.40	77.29	72.37	34.01	**76.45**	**75.33**	**75.13**	63.92
	Recall	83.40	90.58	94.54	88.11	**98.84**	**99.15**	**98.93**	98.08
Test	Accuracy	82.37	82.66	82.34	67.30	**85.27**	**84.51**	**84.70**	81.94
	Precision	78.94	75.42	70.22	35.39	**73.62**	**72.35**	**72.13**	62.30
	Recall	82.61	90.35	94.10	89.17	**97.65**	**98.08**	**97.83**	97.55

Table 3.2 presents a detailed evaluation of different NER models on various entity types related to temporal expressions and events using the i2b2-2012 dataset. The performance is measured using precision (P), recall (R), and F1-score (F1) across four architectures: Standard, Separable, Transposed, and LSTM models. Two variations of BERT models, namely BERT_768 and BERT_128, are evaluated.

For the BERT_768 encoding entity-type performance indicates that the highest F1-score is achieved in the PROBLEM entity under the standard and separable models (0.89 and 0.88 respectively). The DURATION and DATE entities also show strong results, with F1-scores ranging from 0.81 to 0.84 across different models.

However, entities such as EVIDENTIAL and FREQUENCY display relatively lower performance, particularly in recall values. The LSTM model performs poorly across most entity types, yielding an F1-score of 0.00 in multiple cases. This suggests that ConvLSTM may not be sufficient for robust entity recognition in this context.

The BERT_128 variant generally demonstrates improved or competitive performance compared to the BERT_768. The highest F1-score is observed for PROBLEM (0.91) and TEST (0.89). Notably, the separable and standard models perform best for most entity types, outperforming the transposed and ConvLSTM architectures.

For the TIME entity, the highest performance (F1-score = 0.76) is obtained under both the Standard and Separable settings, a significant improvement over the previous model. Similarly, the FREQUENCY entity sees an increase in performance, with the best F1-score (0.71) occurring in the Standard configuration.

The transposed model demonstrates strong recall values, particularly for DATE and OCCURRENCE entities, reaching up to 0.97 and 0.93, respectively. The ConvLSTM model, while still underperforming in some cases, shows better results compared to its counterpart in the BERT_768, achieving an F1-score of 0.83 for TREATMENT and TOKEN entities.

Table 3.2: Detailed test performance by category for events and temporal expressions, evaluated on the i2b2-2012 dataset. While the columns report precision (P), recall (R), and F1-score (F1) metrics for each model.

Entity Type	BERT_768											
	Standard			Separable			Transposed			LSTM		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
TIME	0.70	0.65	0.67	0.68	0.54	0.60	0.79	0.57	0.66	0.00	0.00	0.00
FREQUENCY	0.66	0.42	0.51	0.85	0.34	0.49	0.62	0.45	0.52	0.00	0.00	0.00
DURATION	0.80	0.87	0.84	0.83	0.81	0.82	0.81	0.83	0.82	0.69	0.78	0.73
DATE	0.74	0.90	0.81	0.76	0.85	0.81	0.73	0.92	0.82	0.72	0.71	0.71
EVIDENTIAL	0.51	0.40	0.45	0.67	0.26	0.37	0.96	0.19	0.32	0.00	0.00	0.00
CLINICAL_DEPT	0.90	0.73	0.81	0.93	0.77	0.85	0.87	0.81	0.84	0.00	0.00	0.00
TEST	0.82	0.87	0.84	0.92	0.80	0.85	0.90	0.82	0.85	0.58	0.62	0.60
OCCURRENCE	0.69	0.71	0.70	0.74	0.70	0.72	0.72	0.68	0.70	0.00	0.00	0.00
TREATMENT	0.93	0.75	0.83	0.81	0.85	0.83	0.86	0.81	0.83	0.56	0.73	0.64
PROBLEM	0.95	0.83	0.89	0.96	0.81	0.88	0.90	0.86	0.88	0.85	0.78	0.82
TOKEN	0.78	0.86	0.82	0.78	0.88	0.82	0.79	0.84	0.81	0.65	0.80	0.72
Entity Type	BERT_128											
	Standard			Separable			Transposed			LSTM		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
TIME	0.91	0.65	0.76	**0.92**	**0.65**	**0.76**	0.63	0.80	0.70	0.82	0.50	0.62
FREQUENCY	**0.68**	**0.73**	**0.71**	**0.57**	**0.73**	**0.65**	0.50	0.81	0.62	0.53	0.61	0.57
DURATION	0.86	0.88	0.87	0.83	0.89	0.86	0.76	0.95	0.84	0.76	0.91	0.83
DATE	0.72	0.96	0.83	0.73	0.94	0.82	0.70	**0.97**	**0.82**	0.73	0.93	0.82
EVIDENTIAL	0.80	0.29	0.43	0.59	0.44	0.50	0.40	**0.87**	**0.55**	0.39	0.65	0.48
CLINICAL_DEPT	0.87	0.90	0.88	**0.96**	**0.83**	**0.89**	0.80	0.96	0.88	0.91	0.80	0.85
TEST	**0.97**	**0.83**	**0.89**	0.78	0.98	0.87	0.80	0.97	0.87	0.85	0.86	0.85
OCCURRENCE	0.89	0.68	**0.77**	0.84	0.70	0.76	0.62	**0.93**	0.74	0.77	0.64	0.70
TREATMENT	0.97	0.77	**0.86**	0.95	0.77	**0.85**	0.75	0.97	0.85	0.83	0.83	0.83
PROBLEM	0.98	0.84	**0.91**	0.97	0.85	**0.91**	0.81	0.98	0.89	0.95	0.81	0.87
TOKEN	0.77	0.92	**0.84**	0.80	0.86	0.83	0.73	0.99	**0.84**	0.79	0.83	0.81

While the standard and separable models generally yield the best results, the transposed model excels in recall for certain entities. The ConvLSTM approach, despite improvements in the BERT_128 model, remains suboptimal for this task.

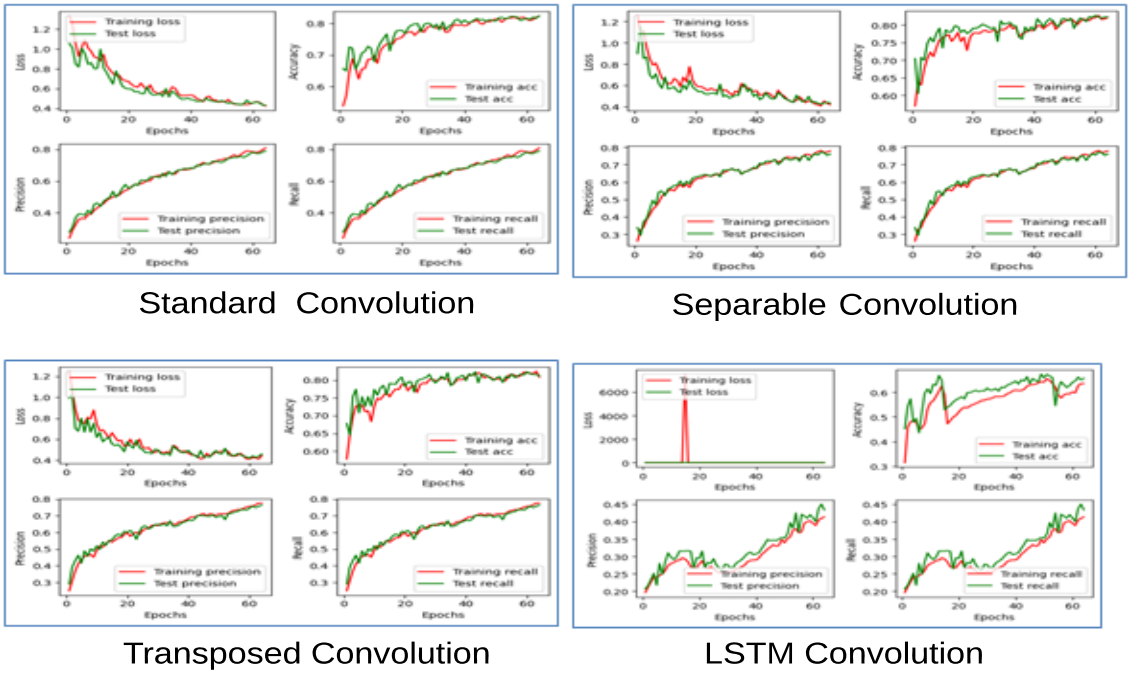


Figure 3.6: Models training progress with BERT_768 encoding.

Figure 3.6 presents a comparative overview of the training behavior of four DL models (standard convolution, separable convolution, transposed convolution, and ConvLSTM) using input representations generated by the BERT-based encoder "BERT_768". For each model, training and validation curves of loss, accuracy, precision, and recall are plotted over 60 epochs to assess learning dynamics and generalization capability. The standard convolution model demonstrates stable and consistent learning progress. The training and test loss decrease steadily over time, with accuracy, precision, and recall improving proportionally. Notably, the close alignment between the training and test curves across all metrics indicates good generalization. This suggests that standard convolutional layers, when combined with 'BERT_768' embeddings, are able to effectively extract task-relevant features from the encoded clinical text. In addition, the separable convolution model, shows slightly improved performance over the standard convolution model, particularly in terms of precision and accuracy. The training and test curves remain closely aligned throughout the learning process, reflecting architectural efficiency and strong generalization. On the other hand, the transposed convolution model also exhibits smooth convergence with low variance between training and test datasets. It achieves competitive precision and recall, indicating that trans-

posed convolution layers can enhance spatial learning without introducing instability. These results suggest that transposed convolution is well-suited for handling structured embeddings like those produced by 'BERT_768', potentially by refining the abstraction of input features at different scales. In contrast, the LSTM convolution model displays training instability, particularly in the early epochs where a sharp spike in loss is observed. While some recovery is apparent in later epochs, fluctuations in precision and recall persist, and a noticeable gap remains between training and test metrics. This behavior indicates poor convergence and weak generalization. One possible explanation is that the temporal modeling capabilities of LSTM layers may not align effectively with the contextual representations provided by the "BERT_768" model, which already captures sequential dependencies at a high level. In summary, both separable convolution and transposed convolution models emerge as the most robust and effective architectures when integrated with transformer-based embeddings, demonstrating high accuracy and consistent generalization. Conversely, LSTM convolution model appears less suitable.

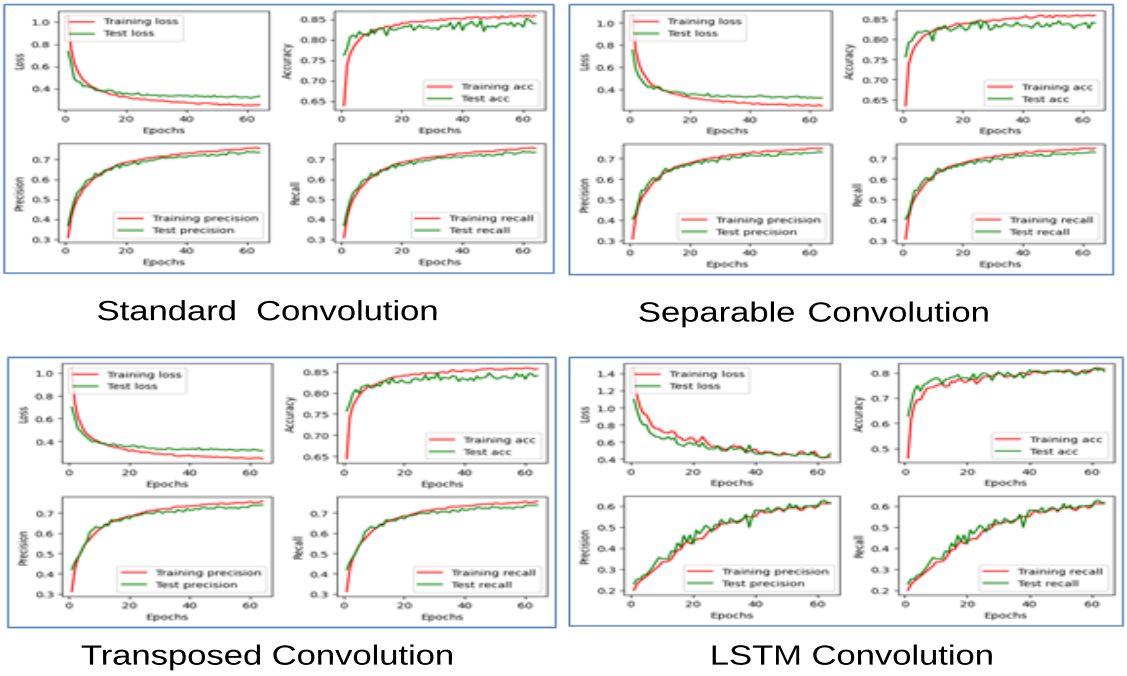


Figure 3.7: Models training progress with BERT_128 encoding.

Figure 3.7 illustrates the training dynamics of four DL architectures (standard convolution, transposed convolution, separable convolution, and ConvLSTM) applied to clinical entity type recognition tasks using BERT_128 embeddings. The Standard Convolution model exhibits a stable and gradual learning trajectory. Its performance improves consistently across epochs, as evidenced by the decreasing loss and increasing accuracy. In contrast, the

LSTM Convolution model displays considerable instability during training. Not only does the loss curve fluctuate, particularly in the early epochs, but the precision and recall metrics also show irregular patterns. As a result, despite its ability to model temporal dependencies, this architecture appears less compatible with BERT_128 embeddings. On the other hand, the transposed convolution model demonstrates improved robustness and convergence. Its curves indicate smooth learning progress, with precision and recall showing consistent gains throughout training. Notably, the gap between training and test performance remains relatively narrow, suggesting reliable generalization. This implies that transposed convolutions may enhance the feature resolution learned from the BERT_128 encoded input by effectively upsampling and refining the intermediate representations. Furthermore, the separable convolution model performs competitively, often outperforming the standard convolution in terms of accuracy. The learning curves reveal that this model benefits from its architectural efficiency, maintaining low training loss while achieving strong performance metrics.

In conclusion, the comparative results emphasize that transposed and separable convolution architectures are the most effective when used in conjunction with BERT_128 embeddings. Their stable convergence and superior generalization highlight their compatibility with dense contextual representations. Conversely, while ConvLSTM model appears less beneficial. This evaluation underscores the importance of aligning model architecture with the nature of the input representation to optimize performance in clinical entity recognition tasks.

The evaluation of the final model, CONSEPTR, which integrates standard convolution, transposed convolution, and separable convolution models, was conducted on three benchmark datasets: i2b2-2012, i2b2-2010, and i2b2-2018. The model's performance was assessed based on the F1-score and overall accuracy, as summarized in Table 3.3.

The results indicate that the CONSEPTR model achieves superior performance on the i2b2-2012 dataset, with an F1-score of 85.37% and an accuracy of 85%. In comparison, the model's performance slightly declines when applied to the i2b2-2010 dataset, yielding an F1-score of 78.99% and an accuracy of 79%. This difference can be attributed to variations in dataset characteristics, including the complexity of event representations and temporal expressions within the clinical narratives.

The relatively higher performance on i2b2-2012 suggests that the model effectively captures temporal and event-related patterns in this dataset, benefiting from its diverse annotations and structured data distribution. Conversely, the lower scores for i2b2-2010 may indicate greater variability or noise within the dataset, requiring further domain-specific tuning or additional feature extraction techniques.

Overall, the results demonstrate the robustness of the CONSEPTR model in handling

event and temporal expression recognition tasks, with promising generalizability across multiple clinical datasets.

Table 3.3: Test results of the final model, CONSEPTR (using Convolution, Transposed Convolution, and Separable Convolution) for i2b2-2012, i2b2-2010, and i2b2-2018 datasets.

Dataset	F1-score	Accuracy	Entity Type	Precision	Recall	F1-score
i2b2-2012	85.37	85	TIME	0.94	0.67	0.78
			FREQUENCY	0.65	0.75	0.70
			DURATION	0.83	0.92	0.87
			DATE	0.73	0.95	0.83
			EVIDENTIAL	0.86	0.25	0.39
			CLINICAL_DEPT	0.95	0.83	0.88
			TEST	0.92	0.86	0.89
			OCCURRENCE	0.90	0.67	0.77
			TREATMENT	0.97	0.78	0.86
			PROBLEM	0.98	0.85	0.91
			TOKEN	0.78	0.92	0.84
i2b2-2010	78.99	79	TEST	0.95	0.74	0.83
			TREATMENT	0.93	0.75	0.83
			PROBLEM	0.95	0.84	0.89
i2b2-2018	38.46	38	FREQUENCY	0.99	0.50	0.67
			DURATION	0.36	0.83	0.50
			TREATMENT	0.90	0.32	0.47
			PROBLEM	0.80	0.41	0.54

The CONSEPTR model was evaluated on three benchmark clinical datasets, i2b2-2012, i2b2-2010 and i2b2-2018, for various entity types related to temporal expressions and medical events. Table 3.3 presents the detailed test results in terms of precision, recall, and F1-score.

The evaluation on the i2b2-2012 dataset demonstrates strong performance across multiple entity types, with F1-scores ranging from 0.39 to 0.91. The model achieves the highest F1-score for the *PROBLEM* entity (0.91), indicating its effectiveness in recognizing medical conditions. Similarly, high F1-scores are observed for *TEST* (0.89), *CLINICAL_DEPT* (0.88), and *DURATION* (0.87), suggesting robust generalization in structured medical terminology.

However, performance varies across different entity categories. The *EVIDENTIAL* category exhibits the lowest F1-score (0.39) due to an imbalanced recall (0.25) despite a relatively high precision (0.86). This suggests that the model struggles to capture all occurrences of evidential markers.

When evaluated on the i2b2-2010 dataset, the CONSEPTR model continues to perform well, but the number of entity types is more limited. The *PROBLEM* entity again achieves a

strong F1-score of 0.89, followed by *TEST* and *TREATMENT*, both with F1-scores of 0.83. These results indicate the model's ability to generalize across datasets while maintaining stable performance for core clinical concepts.

A direct comparison between the datasets reveals that the model consistently performs well on the *PROBLEM*, *TEST*, and *TREATMENT* entities across both datasets (i2b2-2012 and i2b2-2010), with minor variations in precision and recall.

The evaluation of the model on the i2b2-2018 dataset reveals varying performance across different clinical entity types. The overall F1-score was 38.46%, with an accuracy of 38%, indicating that while the model demonstrates some discriminative capability, its performance remains modest when applied to complex clinical narratives. For the FREQUENCY entity, the model achieved a remarkably high precision of 0.99, suggesting that its predictions were nearly always correct when it did identify this entity. However, the recall was considerably lower at 50, indicating that only half of the true instances were successfully detected. This imbalance led to an F1-score of 0.67, reflecting strong precision but limited sensitivity. In contrast, the DURATION entity showed the opposite trend. The model reached a high recall of 0.83, successfully retrieving most of the relevant instances. Yet, its precision was relatively low at 0.36, indicating a tendency to produce false positives. The resulting F1-score of 0.50 reveals a moderate overall performance with a bias toward recall. For the TREATMENT entity, precision was again high (0.90), but recall dropped significantly to 0.32. This indicates that while the model was highly accurate when it predicted a treatment, it failed to identify a large number of actual instances. The corresponding F1-score of 0.47 underscores this lack of coverage. Similarly, the PROBLEM entity yielded a precision of 0.80 and a recall of 0.41, resulting in an F1-score of 0.54.

In summary, although the model exhibits varying precision and recall across entity types, its ability to extract meaningful information from the i2b2-2018 dataset (despite being trained solely on i2b2-2012) demonstrates promising generalization capability. This suggests that the model can transfer learned representations across datasets with different structural characteristics, highlighting its potential utility in real-world clinical text processing. The overall results indicate that CONSEPTR effectively identifies medical entities and temporal expressions, achieving high accuracy for core clinical concepts. However, challenges remain in improving recall for categories such as *EVIDENTIAL* and certain temporal expressions.

3.2.6 Prediction Process

Figure 3.5 provides an overview of the methodology for model training, evaluation, and result prediction. The extraction of entities from textual data plays a crucial role in this process.

Advanced NLP techniques are utilized to enhance entity recognition, specifically through the implementation of the SpaCy library with the *en_core_web_lg* model.

The integration of SpaCy pipeline functions, including *merge_entities* and *merge_noun_chunks*, significantly improves the accuracy and coherence of the entity extraction system by leveraging the linguistic and contextual knowledge encapsulated within the *en_core_web_lg* model. Additionally, to refine entity extraction in the complex domain of medical text, rule-based strategies are incorporated. These rules capture domain-specific patterns and syntactic structures that may not be fully recognized by the language model. This approach extends beyond basic tokenization, enabling the intelligent consolidation and merging of entities within the processed text.

The CONSEPTR model is employed to predict entity types, which applies a majority voting technique.

This approach aimed at capturing temporal and medical knowledge from medical records. Using BERT facilitates the extraction of time-related events and temporal expressions in the text. In addition, different types of CNN models, such as standard convolution, transposed convolution, and separable convolution, are incorporated based on the diverse outcomes these models achieve. In addition the results demonstrate the robustness of the CONSEPTR model in handling event and temporal expression recognition tasks, with promising generalizability across multiple clinical datasets. Furthermore, the experimental findings indicate that the model's performance may be influenced by specific characteristics inherent to medical or clinical datasets, how data is sampled for training and testing, and the chosen vectorization model.

3.3 An advanced approach using Ontologies and DL for Events and TIMEX extraction

3.3.1 Ontologies Description

Since the datasets contain specific types of entities, the DOID, SYMP, and DRON ontologies were used during testing. In the prediction phase, additional ontologies (Food, Disease Drivers, and Uber-anatomy) were incorporated alongside DOID, SYMP, and DRON.

- **The Disease Ontology (DOID or DO):** DOID ontology is an authoritative ontology dedicated to human diseases. It aims to provide the scientific community with coherent, reusable, and enduring descriptions of human disease terms, phenotype traits, and related medical lexicon concepts. These comprehensive definitions result from col-

laborative efforts by biomedical experts coordinated by the University of Maryland School of Medicine's Institute for Genome Sciences. DOID is a practical tool, particularly in genomics, where it annotates disease-related genes and variants, thereby facilitating the interpretation and accessibility of genomic data. Our methodology involves cross-referencing DOID terms with Medical Subject Headings[4], International Classification of Diseases[5], the National Cancer Institute's Thesaurus[6], Systematized Nomenclature of Medicine Clinical Terms[7], and the Online Mendelian Inheritance in Man[8]. By integrating these resources, the disease ontology effectively consolidates various disease and medical terminologies. The primary objective of DOID is to provide an open-source ontology that serves as a genomic resource for integrating biomedical data related to human diseases, disease attributes, and mechanisms, forming a foundational framework for comprehensive biomedical data analysis and enhancing the disease information landscape.

- **The Symptom Ontology (SYMP):** The SYMP defines a symptom as "an observed alteration in function, feeling, or appearance communicated by a patient as an indication of an illness." Recognizing the close relationship between signs and symptoms (where signs represent the objective assessment of disease Symptoms). SYMP thus provides a detailed and structured representation of symptoms to facilitate their accurate annotation and integration into medical data systems.

- **The Drug Ontology (DRON):** DRON plays a crucial role in streamlining the work of researchers on comparative effectiveness and health services. DRON enables efficient querying of National Drug Codes[9], which classify products based on their ingredients, molecular disposition (e.g., beta-adrenergic receptor blockers), therapeutic disposition (e.g., antihypertensives), and physiological effects (e.g., diuretics). The ontology employs the drug terminology of RxNorm [180], maintained by Chemical Entities of Biological Interest (ChEBI) and the National Library of Medicine. By leveraging these standardized terminologies, DRON facilitates the precise categorization and retrieval of drug-related information, supporting advanced research and clinical applications.

- **Food Ontology (FoodOn):** FoodOn is a collaborative project to create a comprehensive and accessible global food ontology that spans the entire farm-to-fork journey.

[4] https://www.nlm.nih.gov/mesh/meshhome.html
[5] https://www.who.int/classifications/icd/en/
[6] https://ncit.nci.nih.gov/ncitbrowser
[7] https://www.snomed.org/snomed-ct
[8] https://www.omim.org/
[9] https://www.fda.gov/drugs/drug-approvals-and-databases/national-drug-code-directory

It accurately and consistently categorizes foods recognized in diverse cultures worldwide[10]. For detailed information about this ontology, see[11].

- **Disease Drivers (DISDRIV):** Human diseases are driven by factors such as environmental, maternal, and social exposures [12].

- **Uber-anatomy ontology (UBERON):** Uberon is a comprehensive cross-species anatomy ontology representing various entities based on traditional anatomical criteria, including structure, function, and developmental lineage. It encompasses extensive relationships with taxon-specific anatomical ontologies, facilitating the integration of functional, phenotype, and expression data[13].

3.3.2 Key Biomedical and Clinical Entity Types

The following section outlines key categories of biomedical concepts that are commonly encountered in medical literature and patient narratives. These categories serve as foundational elements for semantic annotation and knowledge representation.

1. **Anatomy:** Refers to parts of the human body—organs, tissues, and body structures where a condition might originate or be observed.
   Example:
   - Lung (related to respiratory diseases like asthma or tuberculosis)
2. **Disease Drivers (Environmental or Genetic Contributors):** These are factors that contribute to the development of diseases. They can be environmental (e.g., pollution, smoking) or genetic (e.g., inherited mutations).
   Examples:
   - BRCA1 mutation (genetic contributor to breast cancer)
   - Smoking (environmental contributor to lung cancer)
3. **OMIM Susceptibility:** Refers to conditions or genes listed in OMIM (Online Mendelian Inheritance in Man), which catalog genetically inherited diseases and the genes associated with them.
   Example:
   - OMIM:113705 —Susceptibility to Alzheimer's disease.

[10] https://www.ebi.ac.uk/ols4/ontologies/foodon
[11] https://foodon.org/
[12] https://www.ebi.ac.uk/ols4/ontologies/disdriv
[13] https://www.ebi.ac.uk/ols4/ontologies/uberon

4. **Onset (Typical Age of Manifestation) or Phenotype:**
   - **Onset:** Refers to the usual age range when symptoms begin.
   - **Phenotype:** Describes observable characteristics or symptoms of a disease.

   Examples:
   - Early-onset Alzheimer's (onset)
   - Short stature and facial dysmorphism (phenotype of Noonan Syndrome)
5. **Sequence, Disease, Drug, Food Material**
   - **Sequence:** DNA/RNA/protein sequences
   - **Disease:** Diagnosed medical conditions
   - **Drug:** Pharmacological substances used for treatment or prevention
   - **Symptom** refers to physical or mental features indicating a condition, especially as perceived by the patient.
   - **Food Material:** Edibles that can affect health

   Examples:
   - Sequence: ATCGTAGC...
   - Disease: Diabetes
   - Drug: Metformin
   - Symptom: Fatigue
   - Food Material: Gluten
6. **Transmission Process (Pathogen Transmission Mechanism):** How an infectious disease spreads from one host to another.
   Examples:
   - Airborne transmission (e.g., flu, COVID-19)
   - Vector-borne transmission (e.g., malaria via mosquito)

3.3.3 Ontologies using Jena API and Deep Learning

Ontology Reasoning: A principled approach is employed to verify the presence of entities within ontologies, utilizing semantic constructs for accurate classification. This methodology leverages specific queries to determine entity classifications based on ontology properties. The Jena API, a comprehensive Java framework designed for semantic web and linked data applications, is utilized.

SPARQL (SPARQL Protocol and RDF Query Language) serves as the core mechanism for querying, facilitating precise retrieval and classification of entities based on predefined

ontology properties. The following principle is applied to verify the existence of an entity within the ontology:

Where, $\equiv$: Two classes may be expressed to be equivalent. $\exists$: Denotes existential quantification, and $\sqsubseteq$: Denotes subsumption hierarchies for classes and properties.

$$\text{Disease} \equiv \text{hasOBONamespace.}\{\text{disease_ontology}\} \tag{5}$$

$$\text{Symptom} \equiv \exists\text{hasOBONamespace.}\{\text{Symptoms}\} \tag{6}$$

$$\text{Treatment} \equiv \exists\text{hasOBONamespace.}\{\text{chebi_ontology}\} \tag{7}$$

Equation (5) defines the classification of an entity as a "Disease" if it possesses the property "hasOBONamespace" with the value "disease_ontology." This logical entailment enhances the efficiency of disease identification within the ontology.

For the SYMP, Equation (6) establishes that an entity is categorized as a "Symptom" when it has the "hasOBONamespace" property set to "Symptoms."

Similarly, for the DRON, Equation (7) specifies that an entity is classified as a "Treatment" if it has the "hasOBONamespace" property assigned the value "chebi_ontology."

The SPARQL queries utilized in this study facilitate the classification of entities based on their ontology properties. These queries are designed to retrieve entities with specific "hasOBONamespace" values, ensuring accurate identification and categorization. The following SPARQL queries illustrate the approach used:

- **For DOID or DO Ontology** The following SPARQL query retrieves entity t classified as "Disease" within the ontology where the "hasOBONamespace" property is defined as "disease_ontology". Such queries facilitate efficient ontology reasoning and enhance the systematic organization of biomedical data.

```
SPARQL query =
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> +
PREFIX obo: <http://purl.obolibrary.org/obo/> +
PREFIX oboInOwl: <http://www.geneontology.org/formats/oboInOwl#> +
SELECT ?entity WHERE { +
        ?entity rdfs:label ?label . +
        ?entity oboInOwl:hasOBONamespace 'disease_ontology' . +
};
```

- **For SYMP Ontology** This SPARQL query accurately locates and retrieves the entity t that is classified as a symptom within the ontology, ensuring that it meets the criteria for label matching and proper association with the designated ontology namespace.

```
SPARQL query =
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> +
PREFIX obo: <http://purl.obolibrary.org/obo/> +
PREFIX oboInOwl: <http://www.geneontology.org/formats/oboInOwl#> +
SELECT ?entity WHERE { +
        ?entity rdfs:label ?label . +
        FILTER ( LCASE(str(?label)) = \" + t + \") . +
        ?entity oboInOwl:hasOBONamespace 'symptoms' . +
};
```

- **For DRON Ontology** This SPARQL query fetches entity t categorized as "Treatment" in the ontology, where the property "hasOBONamespace" is specified as "chebi_ontology".

```
SPARQL query =
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> +
PREFIX obo: <http://purl.obolibrary.org/obo/> +
PREFIX oboInOwl: <http://www.geneontology.org/formats/oboInOwl#> +
SELECT ?entity WHERE { +
        ?entity rdfs:label ?label . +
        FILTER (LCASE(str(?label)) = \" + t + \") . +
        ?entity oboInOwl:hasOBONamespace 'chebi_ontology' . +
};
```

In addition to the ontologies used at the test level, which correspond to the entities of the datasets at the prediction level, the following ontologies are also utilized:

- **For UBERON ontology** This SPARQL query retrieves entity t classified as "Anatomy" in the ontology, with the "hasOBONamespace" property set to "uberon".

```
SPARQL query =
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> +
PREFIX obo: <http://purl.obolibrary.org/obo/> +
PREFIX oboInOwl: <http://www.geneontology.org/formats/oboInOwl#> +
SELECT ?entity WHERE { +
        ?entity rdfs:label ?label . +
        FILTER ( LCASE(str(?label)) = \" + t + \") . +
        ?entity oboInOwl:hasOBONamespace 'uberon' . +
};
```

- **For DISDRIV ontology** This SPARQL query retrieves entities classified as "Disease Drivers" in the DISDRIV ontology.

```
SPARQL query =
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> +
SELECT ?entity WHERE { +
        ?entity rdfs:label ?label . +
        FILTER ( LCASE(str(?label)) = \" + t + \") . +
};
```

- **For FooDon ontology** The same SPARQL query used for the DISDRIV ontology is also used for the Food ontology.

Equations (8), (9), (10), (11), and (12) illustrate the descriptive logic of the SPARQL queries.

$$\text{Entity} \sqcap (\exists \text{Label}.\{\text{ent}\}) \sqcap (\exists \text{hasOBONamespace}.\text{'symptoms'}) \tag{8}$$

$$\text{Entity} \sqcap (\exists \text{Label}.\{\text{ent}\}) \sqcap (\exists \text{hasOBONamespace}.\text{'disease_ontology'}) \tag{9}$$

$$\text{Entity} \sqcap (\exists \text{Label}.\{\text{ent}\}) \sqcap (\exists \text{hasOBONamespace}.\text{'chebi_ontology'}) \tag{10}$$

$$\text{Entity} \sqcap (\exists \text{Label}.\{\text{ent}\}) \sqcap (\exists \text{hasOBONamespace}.\text{'unberon'}) \tag{11}$$

$$\text{Entity} \sqcap (\exists \text{Label}.\{\text{ent}\}) \tag{12}$$

Here, $\sqcap$ represents logical conjunction (AND).

The integration of these methodologies and technologies significantly enhances the accuracy and efficiency of entity classification in narrative medical texts. Ontological structures are utilized to advance knowledge integration and semantic understanding in healthcare applications. In this regard, Figure 3.8 is crucial for understanding the hierarchical structure of the disease ontology, which is fundamental to achieving precise medical texts classification and effective data integration in the medical domain.

The detailed information for the 'disease' class (particularly on the right side of Figure 3.8) are of great importance. These information include labels, identifiers, and namespaces associated with the disease class, as well as definitions, equivalent classes, subclass relationships, and general class axioms. This comprehensive information significantly improves the accuracy of medical text classification and supports robust ontology-based data integration.

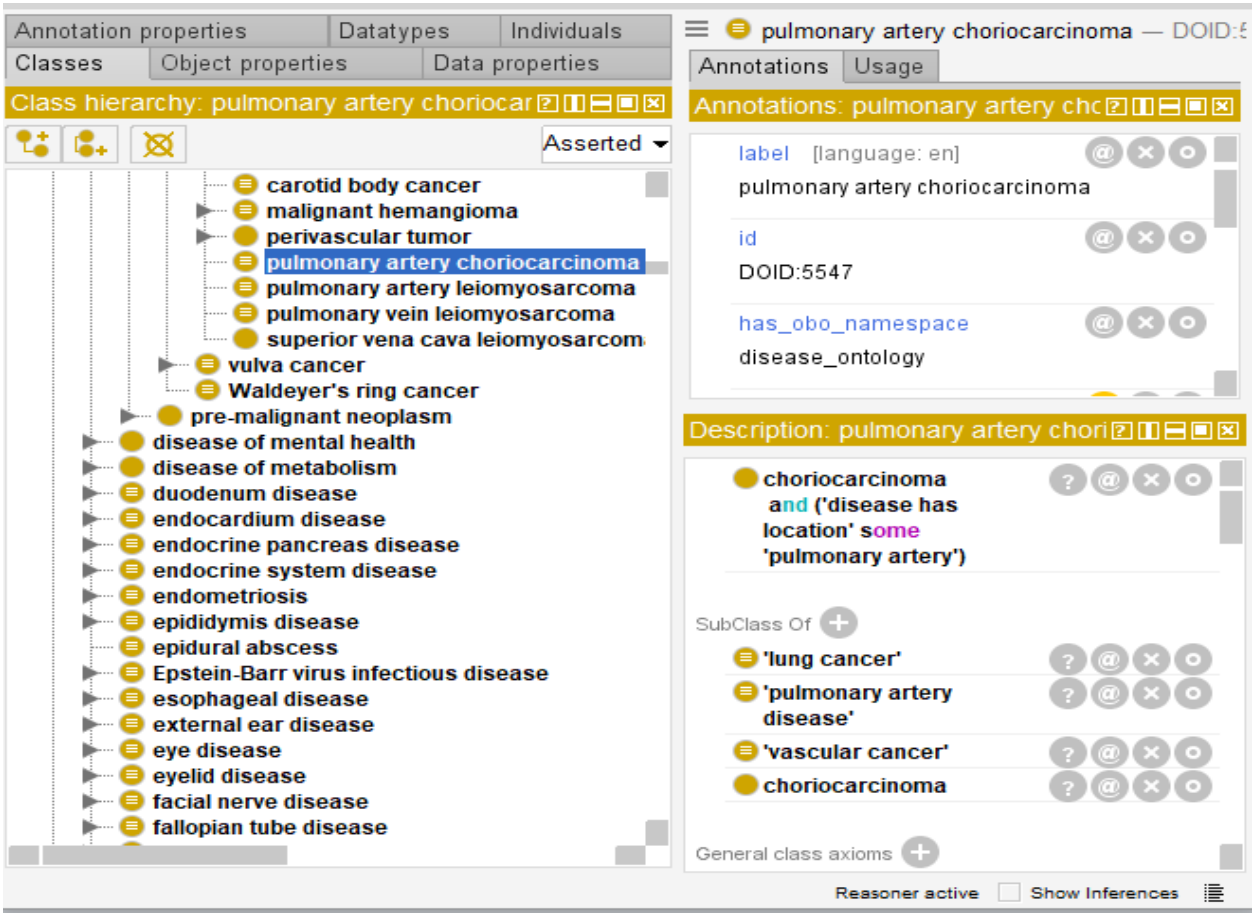


Figure 3.8: A comprehensive view of the DOID as displayed in the Protégé ontology editor, including annotations such as labels, IDs, and namespaces associated with the 'disease_ontology' disease class.

Applied Methods: The structured methodology presented in "Pseudo Algorithm 2" outlines the test performance using the Jena API in combination with DL techniques. By tailoring the processing strategy to the specific attributes of each dataset, the approach ensures robust model performance and reliability in extracting and categorizing medical entities from narrative texts.

The pseudo-algorithm 3 for prediction is designed to extract and classify a range of entities, including events and temporal expressions, from medical narrative texts. It leverages advanced NLP techniques by utilizing the spaCy library for accurate entity recognition.

The process begins with the input of medical narrative text, which is then processed to identify and extract relevant entities. The resulting extraction and classification outputs are visualized using spaCy's powerful visualization tools, and the annotated results are formatted into an XML file, as demonstrated in Figure 3.13. This XML annotation provides detailed metadata for each entity, including its type and its position within the text.

Furthermore, the Jena API is employed for ontology management to enhance the prediction algorithm, while SPARQL queries are utilized for precise data retrieval. This comprehensive approach (integrating NLP techniques, the spaCy library, the Jena API, and SPARQL queries) substantially improves the precision and efficiency of entity extraction and classification in medical narratives. To ensure the proposed approach is robust, performs well, and generalizes effectively, ontologies were integrated with DL, and comprehensive testing was conducted using three datasets with distinct annotation objectives: i2b2-2010, i2b2-2012, and i2b2-2018.

The i2b2-2010 dataset concentrating on concept extraction by identifying medical concepts such as problems, tests, and treatments from patient reports. In contrast, the i2b2-2018 (Track 2) dataset is specifically designed for extracting Adverse Drug Events (ADE) and medication information. In this dataset, entities such as drug, duration, frequency, and reason are utilized, with ”reason” being treated as synonymous with ”problem.”

Algorithm 2 Test's performance using Jena API and DL

Require: *test_data*: Test dataset of i2b2_2012, i2b2_2010, and i2b2_2018
Require: *ontologies*: {DOID, DRON, SYMP}
Ensure: Performance Results using Jena API

```
1: if Using a single ontology then
2:     for each t in test_data do                    ▷ Test the three datasets in test_data
3:         for each onto in ontologies do ▷ onto takes values from {DOID, DRON, SYMP}
4:             for each entity in t do
5:                 if entity ∈ onto then
6:                     if onto ∈ {DOID, SYMP} then
7:                         Type ← "PROBLEM"
8:                     else
9:                         Type ← "TREATMENT"
10:                    end if
11:                else
12:                    Calculate Type using CONSEPTR
13:                end if
14:            end for
15:        end for
16:    end for
17: end if
18: if Utilizing the three ontologies {DOID, DRON, and SYMP} then
19:    for each t in test_data do ▷ This part is applied to the three datasets in test_data
20:        for each entity in t do
21:            if entity ∈ {DOID, SYMP} then
22:                Type ← "PROBLEM"
23:            else if entity ∈ {DRON} then
24:                Type ← "TREATMENT"
25:            else
26:                Calculate Type using CONSEPTR model
27:            end if
28:        end for
29:    end for
30: end if
```

Algorithm 3 Entity prediction via Jena API and DL

Require: Narrative medical text
Ensure: Extracted entities, their types, and annotations

```
Extract entities using spaCy library with "en_core_web_lg", "merge_entities", and "merge_noun_chunks". ▷ Using Jena API - JAVA
for each entity in entities do
    if entity ∈ SYMP then
        Type ← 'Symptom'
    else if entity ∈ DRON then
        Type ← 'Drug'
    else if entity ∈ DOID then
        Type ← 'Disease'
    else if entity ∈ Uber-anatomy then
        Type ← 'Anatomy'
    else if entity ∈ Food then
        Type ← 'Food Material'
    else if entity ∈ Disease Drivers then
        Type ← 'Disease Driver'
    end if
end for
for each entity in entities do
    if entity is not identified by ontologies then
        Calculate Type using CONSEPTR model
    end if
end for
Edit the results using Span and displacy of spaCy
Edit the results in annotation_OntoDeep.xml file
```

Results: In this section, we report the performance of our DL model enhanced with medical ontologies using the Jena API across three benchmark clinical corpora: i2b2-2010, i2b2-2012, and i2b2-2018. For each dataset, we present accuracy and F1-score metrics to capture both overall correctness and the balance between precision and recall. Figure 3.9 provides a summary of the percentage results obtained using ontology-based methods implemented via the Jena API across the i2b2 datasets. Specifically, the results are shown for i2b2-2010 in Figure 3.9a, i2b2-2012 in Figure 3.9b, and i2b2-2018 in Figure 3.9c.

The test results summarized in Table 3.4 compare the performance of different models for entity type extraction on the i2b2-2012 dataset. The models evaluated include the baseline CONSEPTR model and its variants integrated DRON, SYMP, DOID, and the combined ontologies (DRON, SYMP, and DOID, denoted as DSD).

For simpler entities such as TIME, FREQUENCY, and DATE, the performance remains consistent across all model configurations, suggesting that ontology integration has little effect on these entity types. In contrast, for more complex entities such as TREATMENT and PROBLEM, the integration of ontologies leads to noticeable differences in performance. For example, the TREATMENT entity achieves the highest precision of 0.97 in several configurations, while the PROBLEM entity shows slightly varied results, with the baseline and combined ontology models yielding competitive F1-scores.

Overall accuracy and F1-scores are similar across models, with the baseline CONSEPTR model and the CONSEPTR+DOID/CONSEPTR+DSD configurations achieving the highest values (85% accuracy and F1-scores ranging from 84.59 to 85.37). This indicates that while some entity types benefit from the incorporation of ontological knowledge, the overall performance remains robust even with minimal integration. Consequently, the results suggest that combining multiple ontologies can enhance the extraction of more nuanced entities without adversely affecting the extraction of simpler ones.

The test results for the i2b2-2010 dataset highlight the significant impact of incorporating medical ontologies (DRON, SYMP, and DOID) on the accuracy and F1-score of entity extraction as shown in (Table 3.5).

For instance, the baseline CONSEPTR model achieved a precision of 95, recall of 74, and an F1-score of 0.83 for the TEST entity type. When enhanced with ontologies, the models demonstrated improved or comparable performance, with F1-scores reaching 0.85. Similar trends were observed for the TREATMENT and PROBLEM entities. Overall, the integration of ontologies raised the accuracy from 79% to 83% and increased the F1-score from 78.99 to values between 82.60 and 83.13, with the highest F1-score of 83.13 attained by the CONSEPTR+SYMP model.

These results compare favorably with previous studies, where a multi-class SVM model

Table 3.4: The test results using the i2b2-2012 dataset are summarized, which presents the test results of entity type extraction performance using various models. These models include CONSEPTR, CONSEPTR with DRON ontology, CONSEPTR with SYMP, CONSEPTR with DOID, and CONSEPTR with the combined ontologies (DRON, SYMP, and DOID). DSD, denote DRON+SYMP+DOID.

Entities types	CONSEPTR			CONSEPTR+DRON			CONSEPTR+SYMP			CONSEPTR+DOID			CONSEPTR+DSD		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
TIME	0.94	0.67	0.78	0.94	0.67	0.78	0.94	0.67	0.78	0.94	0.67	0.78	0.94	0.67	0.78
FREQUENCY	0.65	0.75	0.70	0.67	0.75	0.71	0.65	0.75	0.70	0.65	0.75	0.70	0.65	0.75	0.70
DURATION	0.83	0.92	0.87	0.83	0.92	0.87	0.84	0.92	0.87	0.84	0.91	0.87	0.83	0.92	0.87
DATE	0.73	0.95	0.83	0.73	0.93	0.82	0.73	0.94	0.82	0.73	0.93	0.82	0.73	0.95	0.83
EVIDENTIAL	0.86	0.25	0.39	0.86	0.25	0.39	0.86	0.25	0.39	0.86	0.25	0.39	0.86	0.25	0.39
CLINICAL_DEPT	0.95	0.83	0.88	0.95	0.81	0.87	0.95	0.81	0.88	0.95	0.82	**0.88**	0.95	0.83	0.88
TEST	0.92	0.86	0.89	0.92	0.84	0.88	0.92	0.84	0.88	0.92	0.85	**0.89**	0.92	0.86	0.89
OCCURRENCE	0.90	0.67	0.77	0.90	0.66	0.76	0.90	0.66	0.76	0.90	0.66	0.76	0.90	0.67	0.77
TREATMENT	**0.97**	**0.78**	0.86	0.84	0.78	0.81	**0.97**	0.77	0.86	**0.97**	0.77	**0.86**	**0.97**	0.78	0.86
PROBLEM	**0.98**	**0.85**	0.91	**0.98**	0.83	**0.90**	0.92	0.85	0.88	0.93	0.85	**0.89**	0.97	0.85	0.90
TOKEN	0.78	**0.92**	0.84	**0.77**	0.89	0.83	0.78	0.90	0.83	0.78	**0.91**	**0.84**	0.78	0.91	0.84
Accuracy	**85**			84			84			**85**			**85**		
F1-score	**85.37**			83.93			84.42			**84.59**			**85.23**		

achieved an F1-score of 0.7861 [181] and another approach reached 0.7725 [182]. Notably, our approach (leveraging medical ontologies alongside DL and using the i2b2-2012 dataset exclusively for training) demonstrates robust and effective performance in medical entity extraction tasks.

Table 3.5: The test results using the i2b2-2010 dataset. Notably, our model uses only the i2b2-2012 dataset as a training dataset, demonstrating its robustness and effectiveness in entity extraction tasks results.

Entities types	CONSEPTR			CONSEPTR+DRON			CONSEPTR+SYMP			CONSEPTR+DOID			CONSEPTR+DSD		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
TEST	**0.95**	**0.74**	0.83	0.92	0.79	0.85	0.92	0.80	0.85	0.92	0.79	0.85	0.92	0.78	0.85
TREATMENT	0.93	0.75	0.83	0.91	0.80	0.85	0.91	0.81	0.86	0.91	0.81	0.86	0.91	0.80	0.85
PROBLEM	0.95	0.84	0.89	0.92	0.86	0.89	0.93	0.86	0.89	0.92	0.86	0.89	0.91	0.86	0.89
Accuracy	79			**83**			**83**			**83**			**83**		
F1-score	78.99			**82.60**			**83.13**			**82.89**			**82.60**		

The test results on the i2b2-2018 dataset for the DURATION entity, as presented in Table 3.6, the baseline CONSEPTR model achieves the highest F1-score (0.67) with a precision of 0.99 and a recall of 0.50, indicating strong precision but moderate recall. In contrast, the incorporation of additional ontologies results in slightly lower F1-scores, suggesting that the baseline model effectively captures DURATION entities.

In the FREQUENCY category, the baseline model also shows the best recall (0.83) and F1-score (0.50), with performance remaining largely consistent across other configurations. Minor reductions in precision are observed when ontologies are integrated. The TREATMENT entity demonstrates notable improvements when enriched with the SYMP ontology. Both the CONSEPTR+SYMP and CONSEPTR+DSD models achieve an F1-score of 0.68,

compared to 0.47 for the baseline model, highlighting the substantial benefit of including symptom-related ontological information for this entity type. Similarly, for the PROBLEM entity, the best performance is seen in the CONSEPTR+SYMP and CONSEPTR+DSD cases, each attaining an F1-score of 0.57. The baseline and CONSEPTR+DOID models yield slightly lower F1-scores (0.54), while the CONSEPTR+DRON variant performs significantly worse (F1-score of 0.36), underscoring the variability introduced by different ontological integration.

Overall, the integration of medical ontologies, particularly SYMP and the combined DSD configuration, leads to significant improvements in both overall accuracy and F1-score. The CONSEPTR+SYMP model, for instance, achieves an accuracy of 54 and a F1-score of 53.56, demonstrating the model's robustness in handling the unique entity structure of the i2b2-2018 dataset. These findings highlight the importance of tailored ontological integration for complex entity types, such as TREATMENT and PROBLEM, and validate the efficacy of the proposed approach even when applied to datasets with differing annotation structures. The importance of using medical ontologies becomes evident through nuanced improvements

Table 3.6: The test results using the i2b2-2018 dataset offer a comprehensive evaluation of entity type extraction models, highlighting the impact of medical ontologies (DRON, SYMP, and DOID). It is important to note that the entity structure of i2b2-2018 differs from that of i2b2-2012, particularly for the TREATMENT and PROBLEM categories, even though our model was trained solely on i2b2-2012.

Entities types	CONSEPTR			CONSEPTR+DRON			CONSEPTR+SYMP			CONSEPTR+DOID			CONSEPTR+DSD		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
DURATION	**0.99**	0.50	**0.67**	0.99	0.38	0.55	0.99	0.47	0.64	0.99	0.49	0.66	0.99	0.47	0.63
FREQUENCY	0.36	**0.83**	**0.50**	0.33	0.62	0.43	0.36	0.79	0.50	0.35	0.80	0.49	0.36	0.81	0.50
TREATMENT	**0.90**	0.32	0.47	0.89	0.40	0.55	**0.91**	**0.54**	**0.68**	**0.90**	0.32	0.47	0.91	0.54	**0.68**
PROBLEM	0.80	0.41	0.54	0.26	0.62	0.36	**0.59**	**0.56**	**0.57**	**0.71**	0.43	0.54	0.56	0.57	**0.57**
Accuracy	38			**43**			**54**			38			**53**		
F1-score	38.46			**42.84**			**53.56**			38.43			**53.34**		

across different entity types. Despite challenges in balancing precision and recall, integrating ontological knowledge enhances the adaptability and effectiveness of models in extracting meaningful information from medical texts. This advancement ultimately enhances the accuracy and utility of entity-type extraction in the complex domain of medical text analysis.

Table 3.7 demonstrates that in the study by [93], higher F1-scores were achieved for Event extraction compared to other studies. [84] Reported a superior TIMEX value of 90.03, but a lower F1-score of 76.77 for event extraction. In contrast, the CONSEPTR model and its variants exhibit competitive F1-scores, indicating their effectiveness in both event and TIMEX extraction tasks. Specifically, the CONSEPTR, CONSEPTR+DRON, CONSEPTR+SYMP, and CONSEPTR+DOID models show F1-scores ranging from 82.60

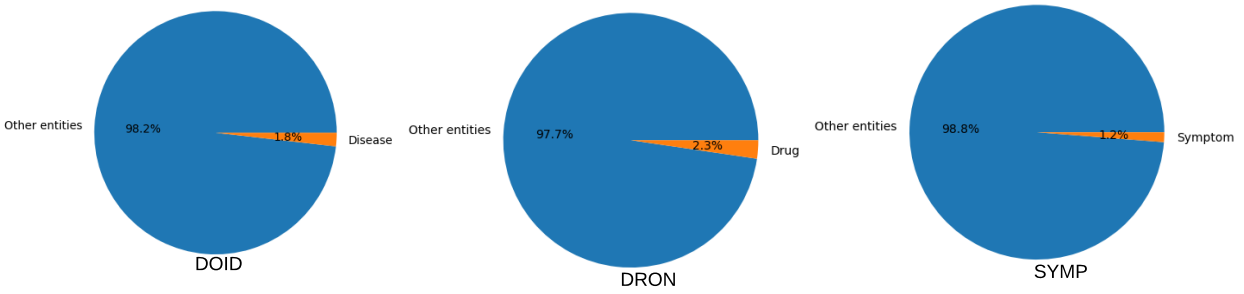
to 85.37, highlighting their robust performance compared to other studies.

By comparing results, we underscore the effectiveness of the hybrid approach, which merges ontologies with DL. This approach covers a broader range of medical concepts and achieves more accurate and context-specific classification, underscoring the pivotal role of ontologies in advancing medical informatics.

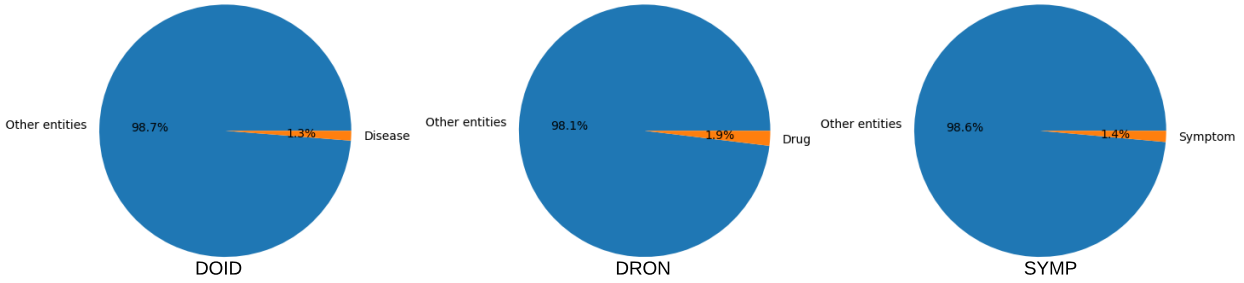
Table 3.7: Comparison of previous studies, used i2b2 dataset, and CONSEPTR-based models in terms of Event and TIMEX extraction, ontology usage, and F1-score performance.

Study	Events	TIMEX	Uses of Ontology	F1-score
[74]	✓			83.5
[93]	✓			90.13
[93]		✓		86.59
[75]	✓			83.07
[75]		✓		83.85
[84]	✓			76.77
[84]		✓		90.03
[80]	✓			87.26
[76]	✓	✓		89.98
[81]	✓			80.91
[94]	✓			78
[94]		✓		77
[183]	✓			80.53
[92]	✓			60.56
[92]		✓		67.8
[152]	✓	✓	UMLS	81.1
[91]	✓	✓	UMLS	79.4
	Our Models			
CONSEPTR	✓	✓		85.37
CONSEPTR + DRON	✓	✓	DRON ontology	83.93
CONSEPTR + SYMP	✓	✓	SYMP ontology	84.42
CONSEPTR + DOID	✓	✓	DOID ontology	82.89
CONSEPTR + DRON + SYMP + DOID	✓	✓	DRON + SYMP + DOID	82.60

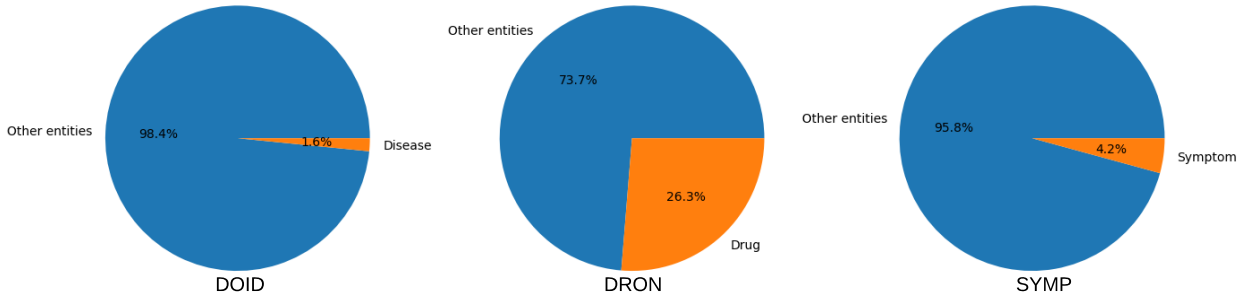
In summary, our findings demonstrate that while the integration of specific medical ontologies can refine entity extraction, the baseline CONSEPTR model already provides competitive performance. This suggests that our approach effectively balances the benefits of ontology integration with the simplicity of a DL framework.



(a) i2b2-2010 dataset.



(b) i2b2-2012 dataset.



(c) i2b2-2018 dataset.

Figure 3.9: The results using ontologies with the Jena API for the i2b2 datasets:Figure 3.9a i2b2-2010,Figure 3.9b i2b2-2012, and Figure 3.9c i2b2-2018.

Prediction Example: We emphasize that in contrast to DL models, when relying on ontologies, "pulmonary artery choriocarcinoma" is classified as a disease (Disease) while "respiratory system" is identified as an anatomical structure (Anatomy). The application of ontologies in the medical domain, as depicted in Figure 3.10 and Figure 3.12 of "pulmonary artery choriocarcinoma" a "respiratory system", presents notable benefits compared to DL models.

The same Note applies to methotrexate which is classified as 'Drugs' by the ontologies, but as 'Treatments' by the CONSEPTR model. "methotrexate" is not identified by DL model (Figure 3.11), but using ontologies is annotate as a "Drug".

This classification doesn't necessarily imply a flaw in our DL model; instead, it reflects how concepts are defined in the i2b2-2012 training dataset. By incorporating ontologies, we can enhance our model by introducing a more comprehensive range of concepts, including those not explicitly covered in the training dataset.

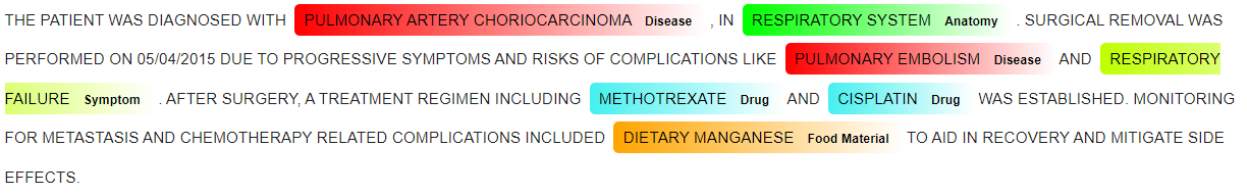


Figure 3.10: Entities extracted using medical ontologies (DOID, SYMP, DRON, Uberanatomy, Food, and Disease Drivers) and Jena API.

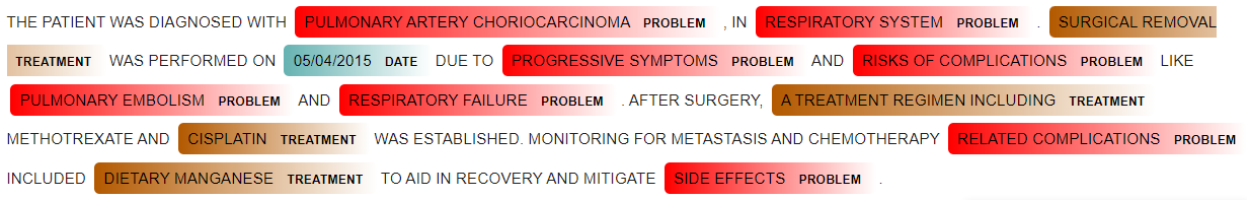


Figure 3.11: Entities are extracted using the CONSEPTR model, which leverages DL. CONSEPTR can identify various events including PROBLEM, TREATMENT, TEST, EVIDENTIAL, CLINICAL_DEPT, and OCCURRENCE. Additionally, it can extract TIMEX such as DATE, TIME, FREQUENCY, and DURATION.

To illustrate, the relationship defined by the ontology see Eq. (13):

$$\text{pulmonary artery choriocarcinoma} \sqcap (\exists \text{hasLocation.respiratory system}) \tag{13}$$

The results provide a detailed hierarchical classification of "pulmonary artery choriocarcinoma" (Figure 3.14).

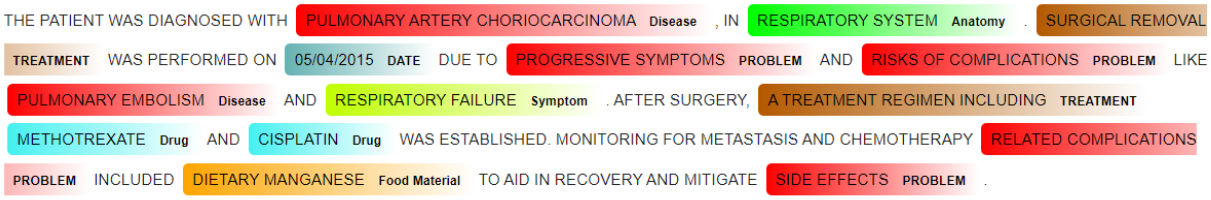


Figure 3.12: Entities extracted using ontologies and DL (CONSEPTR) model.Figure 3.11 shows the performance of our CONSEPTR model in determining entity types. However, our novel approach of combining the model with ontologies using Jena API, which yielded more accurate and precise results. For instance, our model correctly identified "pulmonary artery choriocarcinoma" as a disease rather than a treatment, a distinction that is crucial in the medical field. Similarly, "dietary manganese" was accurately identified as Food Material and "methotrexate" as a drug with greater precision. Even though "dietary manganes"e can generally be considered a treatment and methotrexate is not identified by DL model. The combined model provided more accurate classifications, highlighting the performance of our innovative approach.

It is primarily classified as a type of cancer, specifically a choriocarcinoma. It is associated with the vascular system (vascular cancer, vascular disease) and the respiratory system (respiratory system cancer, respiratory system disease). It is categorized under diseases affecting the lungs (lung cancer, lung disease) and the pulmonary artery (pulmonary artery disease). Some broader categories, like cardiovascular system disease and organ system cancer, encapsulate the condition.
Certain classifications like female reproductive organ cancer and placenta cancer may indicate either overlapping features or data inclusion due to specific ontology structures.
The hierarchical classification highlights the disease's complexity and its interrelation with various body systems, emphasizing the importance of precise and comprehensive categorization in medical ontologies.

Figure 3.14 shows the query result lists several super classes for "pulmonary artery choriocarcinoma" disease in respiratory system, including: artery disease, cardiovascular cancer, cardiovascular system disease, disease of anatomical entity, disease of cellular proliferation, female reproductive organ cancer, female reproductive system disease, lower respiratory tract disease, lung cancer, lung disease, organ system cancer, placenta cancer, placenta disease, pulmonary artery disease, reproductive organ cancer, reproductive system disease, respiratory system cancer, respiratory system disease, uterine cancer, uterine disease, vascular cancer, vascular disease, cancer, choriocarcinoma, and disease.

Where the direct subclasses are as shown inFigure 3.15:

- *lung cancer*

```
<?xml version="1.0" encoding="UTF-8"?>
<ClinicalNarrativeEntityAnnotation>
<TEXT>
<![CDATA[
<The patient was diagnosed with pulmonary artery choriocarcinoma, in respiratory system.
Surgical removal was performed on 05/04/2015 due to progressive symptoms and risks of complications like
pulmonary embolism and respiratory failure. After surgery, a treatment regimen including methotrexate and cisplatin was
established. Monitoring for metastasis and chemotherapy related complications included dietary manganese to aid
in recovery and mitigate side effects. />
]]></TEXT>
<TAGS>
<EVENT id="E0" start="32" end="64" text="PULMONARY ARTERY CHORIOCARCINOMA" type="Disease"/>
<EVENT id="E1" start="69" end="87" text="RESPIRATORY SYSTEM" type="Anatomy"/>
<EVENT id="E2" start="89" end="105" text="SURGICAL REMOVAL" type="TREATMENT"/>
<EVENT id="E3" start="141" end="161" text="PROGRESSIVE SYMPTOMS" type="PROBLEM"/>
<EVENT id="E4" start="166" end="186" text="RISKS OF COMPLICATIONS" type="PROBLEM"/>
<EVENT id="E5" start="194" end="212" text="PULMONARY EMBOLISM" type="Disease"/>
<EVENT id="E6" start="217" end="236" text="RESPIRATORY FAILURE" type="Symptom"/>
<EVENT id="E7" start="253" end="282" text="A TREATMENT REGIMEN INCLUDING" type="TREATMENT"/>
<EVENT id="E8" start="283" end="295" text="METHOTREXATE" type="Drug"/>
<EVENT id="E9" start="300" end="309" text="CISPLATIN" type="Drug"/>
<EVENT id="E10" start="370" end="391" text="RELATED COMPLICATIONS" type="PROBLEM"/>
<EVENT id="E11" start="401" end="418" text="DIETARY MANGANESE" type="Food Material"/>
<EVENT id="E12" start="451" end="463" text="SIDE EFFECTS" type="PROBLEM"/>
<TIMEX id="T0" start="123" end="133" text="05/04/2015" type="DATE"/>
</TAGS>
</ClinicalNarrativeEntityAnnotation>
```

Figure 3.13: Annotated XML file generated using the Jena API and the CONSEPTR model.

- *pulmonary artery disease*
- *vascular cancer*
- *choriocarcinoma*

In conclusion, combining ontologies (using the Jena API and SPARQL) using with DL provides an enhanced level of accuracy and depth in classifying and understanding medical entities. This integration leverages extensive knowledge to generate relevant inferences from data, thereby improving the overall quality of healthcare based on this method. Our proposed approach emphasizes the use of both ontologies and DL. By merging the semantic richness of ontologies, which facilitates the organization and comprehension of complex knowledge within a specific domain, with the computational power of DL, which excels at analyzing and learning from vast amounts of unstructured data, we achieve a powerful synergy. This approach utilizes ontologies for deep contextual understanding and DL for advanced learning and generalization capabilities.

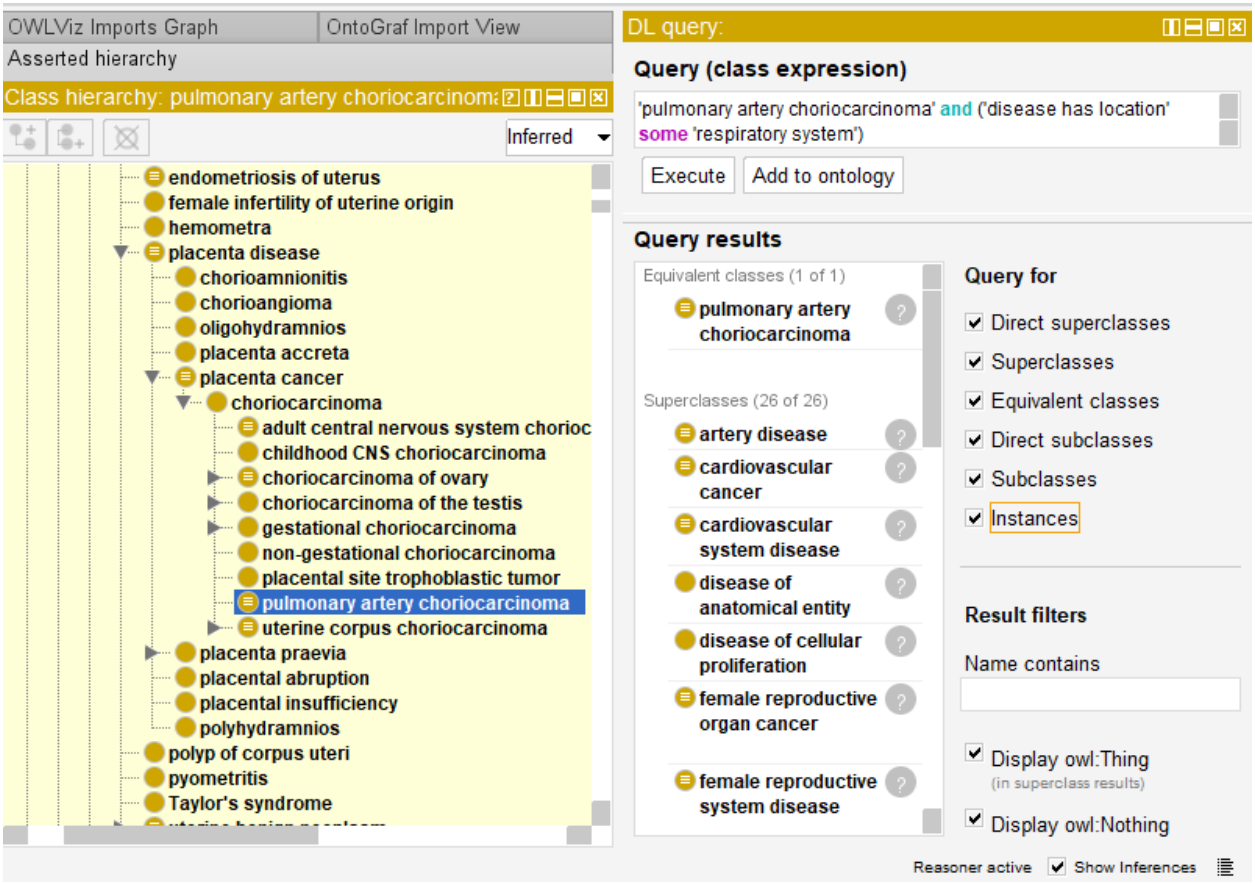


Figure 3.14: Inferences for the entities 'pulmonary artery choriocarcinoma' and 'respiratory system' in the DOID Ontology, using Protégé. This figure illustrates the inferred subclasses generated by the description logic query expression 13 within the DOID ontology, executed in Protégé.

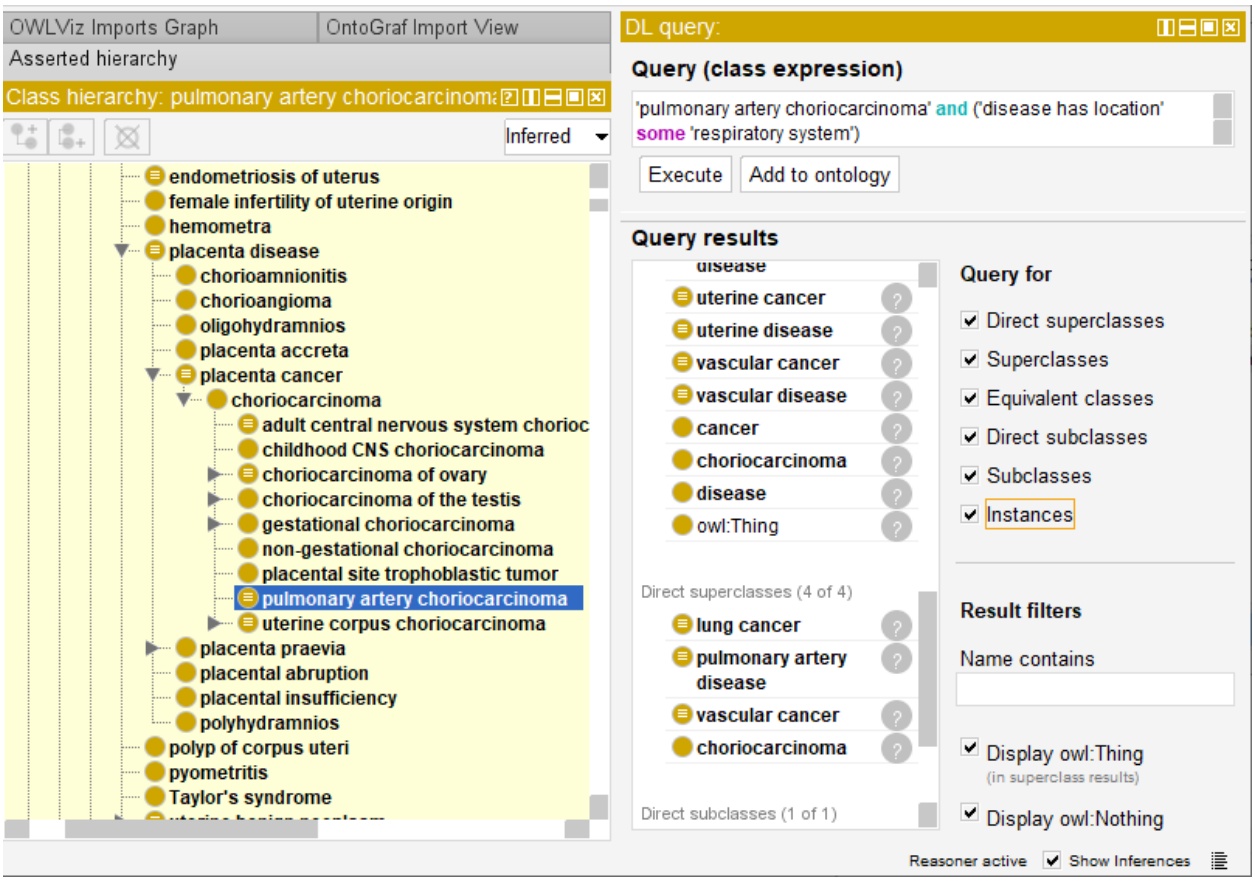


Figure 3.15: Continuation of inferred direct subclasses, from description logic query in Protégé (Figure 3.14).

3.3.4 Ontologies using Owlready2 and DL

This section explores how integrating a DL model (CONSEPTR) with medical ontologies can enhance performance in clinical text analysis [184]. Specifically, we examine the impact of incorporating the DRON, the SYMP, and the DOID ontology using the Owlready2 library[14]. Our goal is to show how combining DL with the rich semantic information provided by medical ontologies can lead to more accurate and meaningful predictions.

Applied Methods: For testing the proposed method using ontologies and DL the applied method follows a two-step approach: first, the entity is searched within the relevant ontology or ontologies (We employed the principles cited in Equations 5, 6, and 7 to check if an entity exists in an ontology). If the entity is not found, the CONSEPTR model is used to determine its type. In this case the ontologies used are DOID, SYMP, and DRON to extract entities as TREATMENT or PROBLEM for testing the proposed method. The CONSEPTR model utilizes a prediction method based on majority voting among standard, transposed, and separable convolution models. This approach enables the extraction of both medical events (EVIDENTIAL, CLINICAL_DEPT, TEST, OCCURRENCE, TREATMENT, and PROBLEM) and temporal expressions (TIME, DURATION, DATE, and FREQUENCY).

For prdediction process Using Owlready2 and the DOID, SYMP, and DRON ontology, the following entities can be extracted: Anatomy, Disease Drivers, OMIM Susceptibility, Onset , Sequence, Disease, Food Material, Transmission Process, Drug, and Symptom.

The entities are first identified using SpaCy's large English model (en_core_web_lg) with the merge_entities and merge_noun_chunks components (as presented in Pseudo Algorithm Prediction 4). Each candidate span is then subjected to a series of ontology lookups to assign a semantic label. Initially, the span is queried against the Disease Ontology (DOID) via a case-insensitive search. A predefined set of ontology identifier patterns—covering anatomical entities (UBERON), drugs (CHEBI), diseases (DOID), disease drivers (DISDRIV), genetic susceptibilities (OMIM), food materials (FOODON), phenotypic features (HP), sequence (SO), symptoms (SYMP), and transmission processes (TRANS)—is used to map any matches directly to their corresponding entity types. If DOID does not yield a match, the algorithm proceeds sequentially to symptom and drug ontologies. A successful lookup in the symptom ontology (SYMP) assigns the label "Symptom," whereas a match in the Drug Ontology (DRON) with CHEBI patterns results in the "Drug" label. Spans that remain unlabeled after these cascaded ontology searches are passed to the CONSEPTR DL model. CONSEPTR applies majority voting over its constituent convolutional architectures

[14] https://owlready2.readthedocs.io/en/v0.47/

Algorithm 4 Pseudo algorithm prediction using Owlready2

Require: Narrative medical text
Ensure: Entities with their types (annotated entities)

Begin
Extract the entities using ("en_core_web_lg", "merge_entities", "merge_noun_chunks") of spacy library.
for `ent` in {entities} **do**
    $r \leftarrow$ DOID.search(label=ent.lower())
    $T \leftarrow \{\{id.\{obo.\text{UBERON*}\}, id.\{obo.\text{CHEBI*}\},$
    $id.\{obo.\text{DOID*}\}, id.\{obo.\text{DISDRIV*}\}, \},$
    $id.\{obo.\text{OMIM*}\}, id.\{obo.\text{FOODON*}\},$
    $id.\{obo.\text{HP*}\}, id.\{obo.\text{SO*}\}, \}\}$
    $id.\{obo.\text{SYMP*}\}, id.\{obo.\text{TRANS*}\}\}$
    if $\exists r \in T$ **then**
        The type of the entity $\in$ ['anatomy', 'Drug', 'Disease', 'disease driver', 'omim_susceptibility', 'food material', 'onset or phenotype', 'sequence','Symptom', 'transmission process']
    else
        $r \leftarrow$ SYMP.search(label=ent.lower())
        if $\exists$ id.{obo.SYMP*} **then**
            label = 'Symptom'
        else
            $r \leftarrow$ DRON.search(label=ent.lower())
            if $\exists$ id.{obo.CHEBI*} **then**
                label = 'Drug'
            else
                Final output = CONSEPTR(ent)
    ▷ CONSEPTR model uses majority voting based on {standard convolutional, transposed convolutional, and separable convolutional models defined in [185]}
            end if
        end if
    end if
end for
Editing of the results using Span and displacy of Spacy
Editing of the results on *XML file*
End

(standard, transposed, and separable convolutions) to predict the entity type when no deterministic ontology match exists. Finally, the raw predictions are refined using SpaCy's visualization tools (displaCy) to correct span boundaries and formatted output is written to an XML file. Figure 3.20 presents the XML file generated for the medical text processed in Section 3.3.4.

Results and Discussion: To assess the robustness, performance, and generalizability of our ontology-enhanced DL approach, we conducted extensive evaluations using the i2b2 2010, 2012, and 2018 datasets—each tailored for distinct annotation tasks.

Table 3.8 highlights the key findings for i2b2-2012 dataset, showing that integrating medical ontologies with the CONSEPTR model produces results nearly equivalent to those of the standalone CONSEPTR model across most entity types. For the TREATMENT entity, both the CONSEPTR+DRON and CONSEPTR+(DRON, SYMP, DOID) models show a drop in recall to 81%, along with a slight decrease in precision to 91%. However, the F1-score remains consistent at 86% across all models. A similar trend is observed for the PROBLEM entity.

Table 3.8: The performance results using i2b2-2012 with owlready2.

Entities types	CONSEPTR			CONSEPTR+DRON			CONSEPTR+SYMP			CONSEPTR+DOID			CONSEPTR+DSD[1]		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
TIME	0.94	0.67	0.78	0.94	0.67	0.78	0.94	0.67	0.78	0.94	0.67	0.78	0.94	0.67	0.78
FREQUENCY	0.65	0.75	0.70	0.65	0.75	0.70	0.65	0.75	0.70	0.65	0.75	0.70	0.65	0.75	0.70
DURATION	0.83	0.92	0.87	0.83	0.92	0.87	0.83	0.92	0.87	0.83	0.92	0.87	0.83	0.92	0.87
DATE	0.73	0.95	0.83	0.73	0.95	0.83	0.73	0.95	0.83	0.73	0.95	0.83	0.73	0.95	0.83
EVIDENTIAL	0.86	0.25	0.39	0.86	0.25	0.39	0.86	0.25	0.39	0.86	0.25	0.39	0.86	0.25	0.39
CLINICAL_DEPT	0.95	0.83	0.88	0.95	0.83	0.88	0.95	0.83	0.88	0.95	0.83	0.88	0.95	0.83	0.88
TEST	0.92	0.86	0.89	0.92	0.86	0.89	0.92	0.86	0.89	0.92	0.86	0.89	0.92	0.86	0.89
OCCURRENCE	0.90	0.67	0.77	0.90	0.67	0.77	0.90	0.67	0.77	0.90	0.67	0.77	0.90	0.67	0.77
TREATMENT	**0.97**	**0.78**	0.86	**0.91**	**0.81**	0.86	0.97	**0.78**	0.86	0.97	**0.78**	0.86	**0.91**	**0.81**	0.86
PROBLEM	**0.98**	**0.85**	0.91	**0.98**	**0.85**	0.91	**0.95**	**0.87**	0.91	**0.96**	**0.86**	0.91	**0.96**	0.86	0.91
TOKEN	0.78	**0.92**	0.84	0.78	**0.90**	0.84	0.78	**0.90**	0.84	0.78	0.91	0.84	0.78	**0.89**	**0.83**
Accuracy	85			85			85			85			85		
F1-score	**85.37**			85.12			**85.31**			85.25			85.00		

[1]**DSD:** DRON+ SYMP+DOID

In Table 3.9, the TEST entity exhibits high precision (95%) across all models, though recall varies slightly. Notably, the CONSEPTR model integrated with DRON or the full combination of ontologies (DRON, SYMP, DOID) records a marginal decrease in recall compared to the baseline CONSEPTR model. For the TREATMENT entity, while precision remains high across models, differences appear in recall and F1-score. The CONSEPTR+(DRON, SYMP, DOID) model achieves the highest F1-score of 85%. Likewise, for the PROBLEM

entity, all models maintain high precision, with the combined model again achieving the top F1-score at 90%.

Overall model accuracy shows a slight upward trend, reaching a peak of 81% with the CONSEPTR+(DRON, SYMP, DOID) configuration. The F1-score, which reflects performance across all entity types, also supports this outcome, with the combined model achieving the highest score of 80.98%.

In summary, integrating CONSEPTR with DRON, SYMP, and DOID ontologies leads to consistently strong results (using i2b2-2010 dataset), particularly in terms of F1-score and overall accuracy. This emphasizes the added value of ontological knowledge in enhancing the performance and generalizability of entity extraction models for clinical texts. Table 3.10

Table 3.9: The performance results using i2b2-2010 with owlready2.

Entities types	CONSEPTR			CONSEPTR+DRON			CONSEPTR+SYMP			CONSEPTR+DOID			CONSEPTR + DSD		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
TEST	**0.95**	**0.74**	0.83	0.95	0.73	0.83	0.95	0.73	0.83	0.95	0.73	0.83	0.95	0.73	0.83
TREATMENT	0.93	0.75	0.83	0.93	**0.79**	**0.85**	0.93	0.75	0.83	0.93	0.75	0.83	0.93	**0.79**	**0.85**
PROBLEM	0.95	0.84	0.89	0.95	0.84	0.89	0.95	**0.85**	**0.90**	0.95	**0.85**	**0.90**	0.95	0.86	0.90
Accuracy	79			80			80			80			**81**		
F1-score	78.99			**79.97**			79.51			79.58			**80.98**		

provides further insight. For the DURATION entity, all models achieve the same precision (99%), while the highest F1-score (66%) is obtained using the CONSEPTR+DOID model. In the FREQUENCY category, while precision remains consistent, the best recall (82%) is achieved by CONSEPTR+DOID. For the PROBLEM entity, CONSEPTR combined with DRON yields a high precision of 80%. Similarly, for TREATMENT, both CONSEPTR+DOID and CONSEPTR+SYMP models record a precision of 90%. These findings reinforce the importance of incorporating medical ontologies to enhance model robustness and adaptability across diverse clinical entity types. The value of incorporating

Table 3.10: The performance results using i2b2-2018 and owlready2.

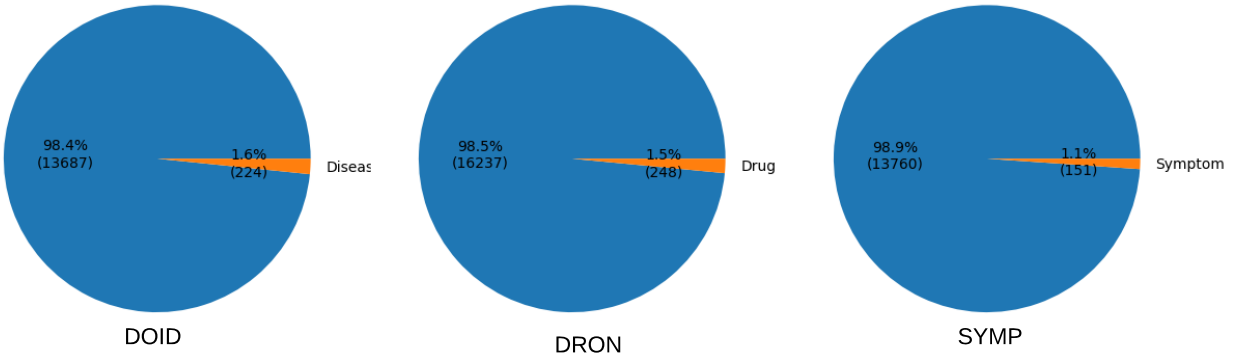
Entities types	CONSEPTR			CONSEPTR+DRON			CONSEPTR+SYMP			CONSEPTR+DOID			CONSEPTR+DSD		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
DURATION	**0.99**	0.50	**0.67**	**0.99**	0.44	0.61	**0.99**	0.49	**0.65**	**0.99**	0.50	**0.66**	**0.99**	0.42	0.59
FREQUENCY	0.36	**0.83**	**0.50**	0.36	0.75	0.49	0.36	**0.81**	**0.50**	0.36	**0.82**	**0.50**	0.36	0.72	0.48
TREATMENT	**0.90**	0.32	0.47	0.79	0.40	**0.53**	**0.90**	0.31	0.46	**0.90**	0.32	0.47	0.78	0.38	0.51
PROBLEM	**0.80**	0.41	**0.54**	**0.80**	0.36	0.50	0.62	0.42	0.50	0.72	0.41	0.52	0.56	0.37	0.45
Accuracy	38			**41**			38			38			40		
F1-score	38.46			**41.36**			37.72			38.14			39.89		

medical ontologies is evident in the nuanced performance gains across different entity types. Although challenges remain (particularly in balancing precision and recall) the integration

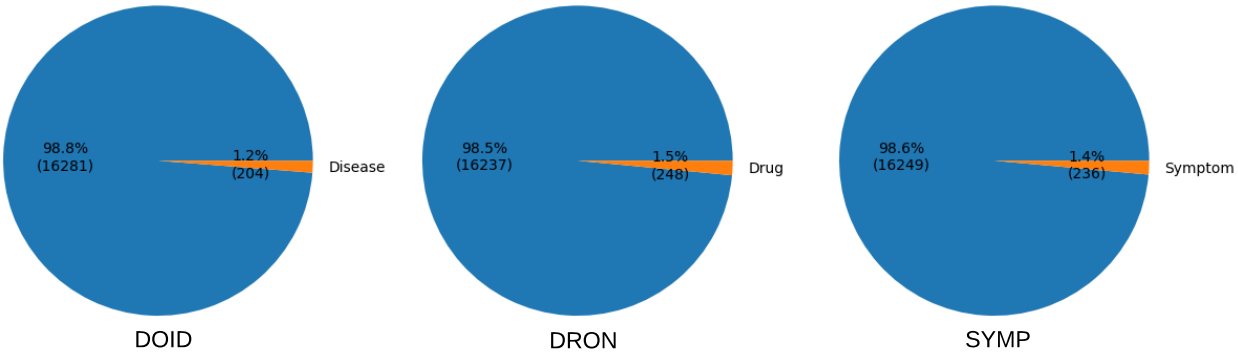
of ontological knowledge enhances the model's adaptability and effectiveness in extracting meaningful information. This ultimately improves both the accuracy and practical utility of entity extraction in the complex field of medical text analysis.

Figure 3.16 summarizes the percentage-based outcomes achieved with ontology-driven methods implemented using Owlready2 across the i2b2 corpora. The detailed results for i2b2-2010, i2b2-2012, and i2b2-2018 are presented in Figures 3.16a, 3.16b, and 3.16c, respectively.

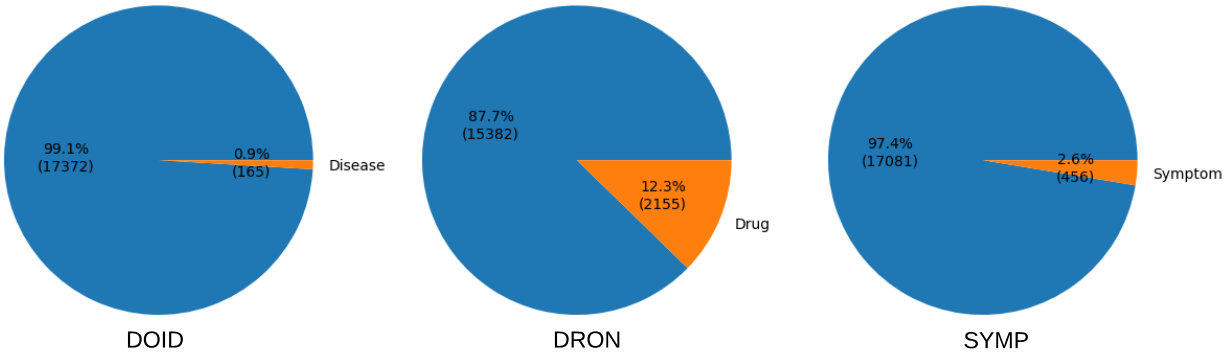
Previous studies have achieved notable F1-scores, such as [93] (90.13 for events, 86.59 for TIMEX), [84] (76.77 for events, 90.03 for TIMEX), and [76] (89.98 for TIMEX and events). Other approaches, including [75] (83.07 for events, 83.85 for TIMEX), [81] (80.91 for events), [80] (87.26 for events), and [74] (83.5 for events), have demonstrated strong results but lacked ontology integration. Although models leveraging UMLS ontologies, such as [152] (81.10) and [91] (79.40), improved semantic representation, they did not fully address the complexity of medical data. Our proposed models achieved competitive F1-scores (approximately 85), demonstrating the efficacy of combining structured domain knowledge with DL. Our method extends entity recognition beyond standard clinical concepts to include anatomy, genetic disease drivers, drug classifications, and transmission processes, enhancing the depth and usability of extracted information.



(a) i2b2-2010 dataset.



(b) i2b2-2012 dataset.



(c) i2b2-2018 dataset.

Figure 3.16: The results using ontologies with the Owlready2 library for the i2b2-2012, i2b2-2010, and i2b2-2018 datasets.

Prediction Example: In this section, we illustrate a concrete example of how medical ontologies combined with DL models generate predictions on clinical narratives. We emphasize that in contrast to DL models, when relying on ontologies, HEART DISEASE is classified as a disease (Disease) while CARDIAC SEPTUM is identified as an anatomical structure (Anatomy). The application of ontologies in the medical domain, as depicted in Figure 3.17 andFigure 3.19 of HEART DISEASE and CARDIAC SEPTUM, presents notable benefits compared to DL models. Indeed, conversely, DL models these concepts, exemplified by HEART DISEASE and CARDIAC SEPTUM being classified as problems "PROBLEM" Figure 3.18. The same Note applies to METOPROLOL and ENALAPRIL, which are classified as 'Drugs' by the ontologies, but as 'Treatments' by the CONSEPTR model. This classification does not necessarily indicate a flaw in our DL model but rather reflects how concepts are delineated in the training dataset (i2b2-2012). Incorporating the DOID can enhance our model by introducing a more comprehensive array of concepts, some of which are not explicitly outlined in the training dataset. DOID, for example, encompasses notions such as 'anatomy', 'disease driver' (Environmental or genetic mechanisms driving the occurrance of complex diseases), 'omim_susceptibility', 'onset' (The age group in which disease manifestations appear) or phenotype', 'sequence', and 'transmission process' (A process that is the means during which the pathogen is transmitted directly or indirectly from its natural reservoir, a susceptible host or source to a new host). This conceptual diversity encompasses a broader scope of medical understanding and improves the extracted information's precision and significance.

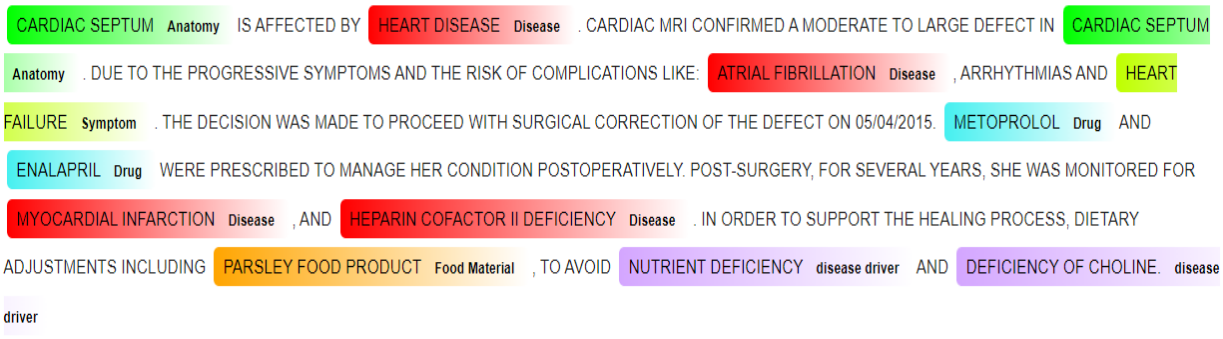


Figure 3.17: Entities extracted using medical ontologies (DOID, SYMP, and DRON).

The utilization of ontologies plays a crucial role in assimilating novel information from identified entities. To illustrate, the relationship defined by the ontology:

$$\text{heartdisease} \sqcap (\exists \text{hasLocation.cardiacseptum}) \tag{11}$$

The inferred result of query 11 is a subset of "atrial heart septal defect". This indicates that

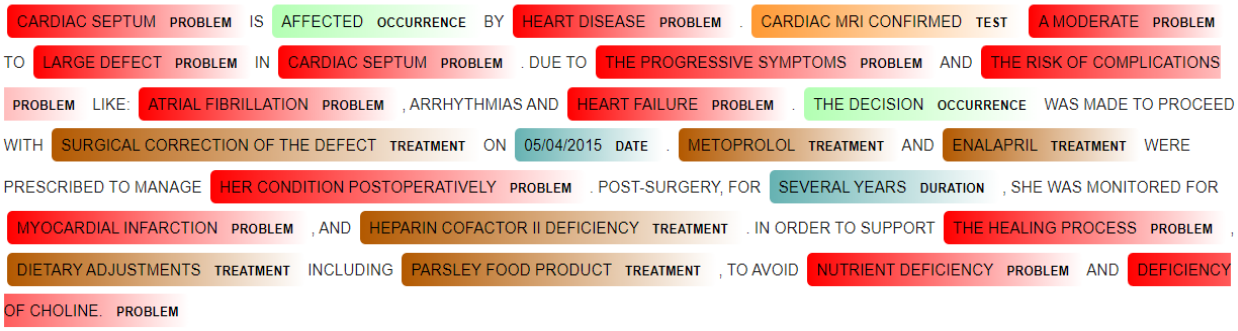


Figure 3.18: Entities extracted using DL (CONSEPTR model). CONSEPTR can extract events such as PROBLEM, TREATMENT, TEST, EVIDENTIAL, CLINICAL_DEPT, and OCCURRENCE. It can also extract TIMEX such as DATE, TIME, FREQUENCY, and DURATION.

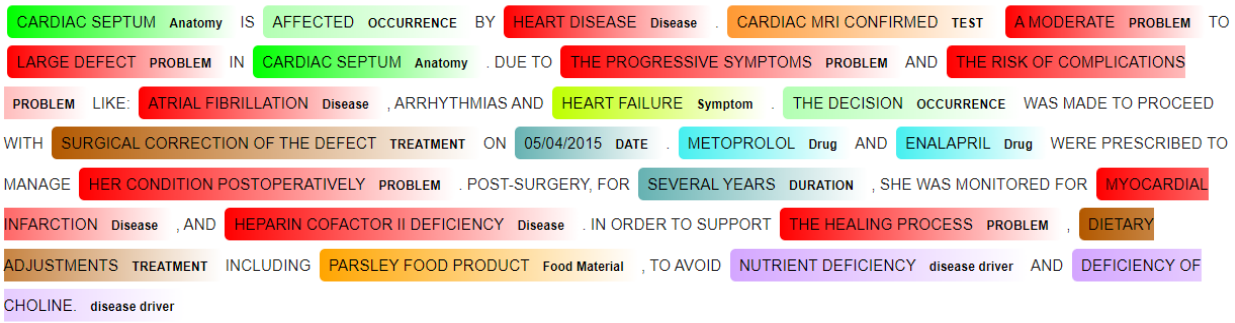


Figure 3.19: Entities extracted using medical ontologies and DL (DRON+SYMP+DOID+CONSEPTR model). Figure 3.18 shows the performance of our CONSEPTR model in determining entity types. However, our novel approach of combining the model with ontologies, which yielded more accurate and precise results. For instance, our model correctly identified HEPARIN COFACTOR II DEFICIENCY as a disease rather than a treatment, a distinction that is crucial in the medical field. Similarly, PARSLEY FOOD PRODUCT was accurately identified as Food Material and NUTRIENT DEFICIENCY as a disease driver with greater precision. Even though PARSLEY FOOD PRODUCT can generally be considered a treatment and NUTRIENT DEFICIENCY a problem, the combined model provided more accurate classifications, showcasing the potential of our innovative approach.

if a patient is diagnosed with heart disease situated at the cardiac septum, it can be deduced that they are afflicted with an atrial heart septal defect. This deductive capacity, facilitated by the ontologies, is pivotal in enabling precise diagnosis and suitable patient care, thereby demonstrating the potential of our research.

In conclusion, intertwining ontologies with DL offers an additional level of accuracy and profundity in classifying and explaining medical entities. It further enables the utilization of vast knowledge and the creation of pertinent inferences from data, thereby enhancing the

```
<?xml version="1.0" encoding="UTF-8"?>
<ClinicalNarrativeEntityAnnotation>
<TEXT><![CDATA[
<cardiac septum is affected by heart disease. Cardiac MRI confirmed a moderate to large defect in cardiac septum.
Due to the progressive symptoms and the risk of complications like: atrial fibrillation, arrhythmias and heart failure. The decision was made to proceed
with surgical correction of the defect on 05/04/2015. metoprolol and enalapril were prescribed to manage her condition postoperatively. Post-surgery,
For several years, she was monitored for myocardial infarction, and heparin cofactor II deficiency.
In order to support the healing process, dietary adjustments including parsley food product, to avoid nutrient deficiency and deficiency of choline.
/]]></TEXT>
<TAGS>
<EVENT id="E0" start="0" end="13" text="CARDIAC SEPTUM" type="Anatomy"/>
<EVENT id="E1" start="19" end="27" text="AFFECTED" type="OCCURRENCE"/>
<EVENT id="E2" start="31" end="44" text="HEART DISEASE" type="Disease"/>
<EVENT id="E3" start="46" end="67" text="CARDIAC MRI CONFIRMED" type="TEST"/>
<EVENT id="E4" start="68" end="78" text="A MODERATE" type="PROBLEM"/>
<EVENT id="E5" start="82" end="94" text="LARGE DEFECT" type="PROBLEM"/>
<EVENT id="E6" start="98" end="112" text="CARDIAC SEPTUM" type="Anatomy"/>
<EVENT id="E7" start="121" end="145" text="THE PROGRESSIVE SYMPTOMS" type="PROBLEM"/>
<EVENT id="E8" start="150" end="173" text="THE RISK OF COMPLICATIONS" type="PROBLEM"/>
<EVENT id="E9" start="182" end="201" text="ATRIAL FIBRILLATION" type="Disease"/>
<EVENT id="E10" start="219" end="232" text="HEART FAILURE" type="Symptom"/>
<EVENT id="E11" start="234" end="246" text="THE DECISION" type="OCCURRENCE"/>
<EVENT id="E12" start="272" end="303" text="SURGICAL CORRECTION OF THE DEFECT" type="TREATMENT"/>
<EVENT id="E13" start="321" end="331" text="METOPROLOL" type="Drug"/>
<EVENT id="E14" start="336" end="345" text="ENALAPRIL" type="Drug"/>
<EVENT id="E15" start="372" end="401" text="HER CONDITION POSTOPERATIVELY" type="PROBLEM"/>
<EVENT id="E16" start="458" end="479" text="MYOCARDIAL INFARCTION" type="Disease"/>
<EVENT id="E17" start="485" end="515" text="HEPARIN COFACTOR II DEFICIENCY" type="Disease"/>
<EVENT id="E18" start="537" end="556" text="THE HEALING PROCESS" type="PROBLEM"/>
<EVENT id="E19" start="558" end="577" text="DIETARY ADJUSTMENTS" type="TREATMENT"/>
<EVENT id="E20" start="588" end="608" text="PARSLEY FOOD PRODUCT" type="Food Material"/>
<EVENT id="E21" start="619" end="638" text="NUTRIENT DEFICIENCY" type="disease driver"/>
<EVENT id="E22" start="643" end="662" text="DEFICIENCY OF CHOLINE" type="disease driver"/>
<TIMEX id="T0" start="309" end="319" text="05/04/2015" type="DATE"/>
<TIMEX id="T1" start="421" end="434" text="SEVERAL YEARS" type="DURATION"/>
</TAGS>
</ClinicalNarrativeEntityAnnotation>
```

Figure 3.20: Annotated XML file generated using the Owlready2 library and the CONSEPTR model.

overall quality of healthcare predicated on this approach.

Figure 3.21 shows that heart septal defect is an equivalent class for the query expression. It means that the class heart septal defect is inferred to be exactly the set of individuals with heart disease located in the cardiac septum. The query result lists several super classes for heart septal defect, including:

- cardiovascular system disease
- congenital heart disease
- disease of anatomical entity
- heart disease
- physical disorder
- disease

Congenital heart disease is identified as a direct superclass of heart septal defect, indicat-

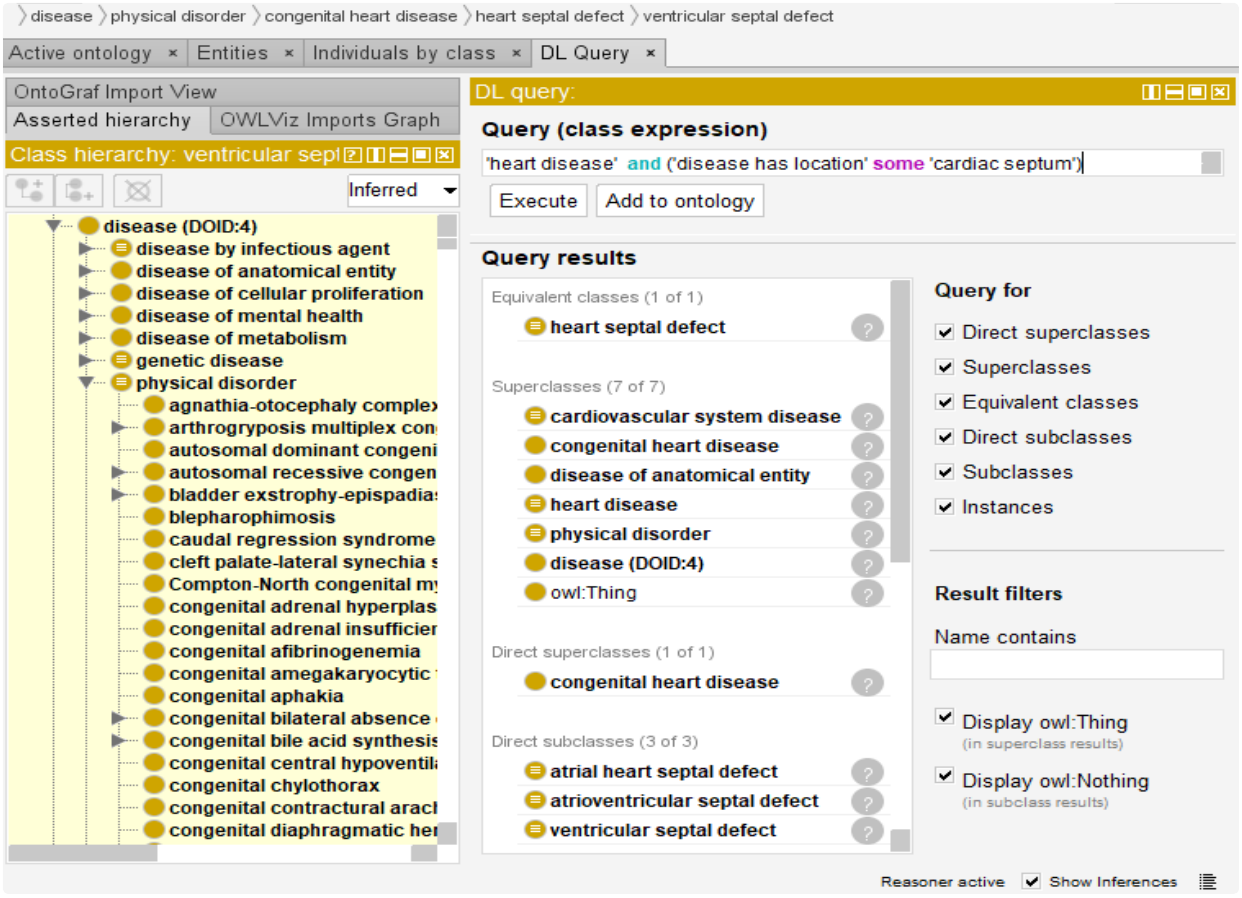


Figure 3.21: Inferences for the entities 'Heart Disease' and 'Cardiac Septum' in DOID Ontology Using Protégé. This figure displays the inferred subclasses resulting from the description logic query expression 11 within the DOID ontology, as executed in Protégé.

ing that heart septal defect is a type of congenital heart disease. Where the direct subclasses of heart septal defect are shown as (Figure 3.22):

- atrial heart septal defect
- atrioventricular septal defect
- ventricular septal defect

The ontology's reasoner infers that these specific defects are heart septal defects. Furthermore, based on the information in Figure 3.21, the ontology and the reasoner used in Protégé (Pellet) have inferred the hierarchical relationships and equivalences for the class heart septal defect based on the provided query. The class heart septal defect is equivalent to the query expression, which means it satisfies both conditions

- It is a type of heart disease.
- It has a location in the cardiac septum.

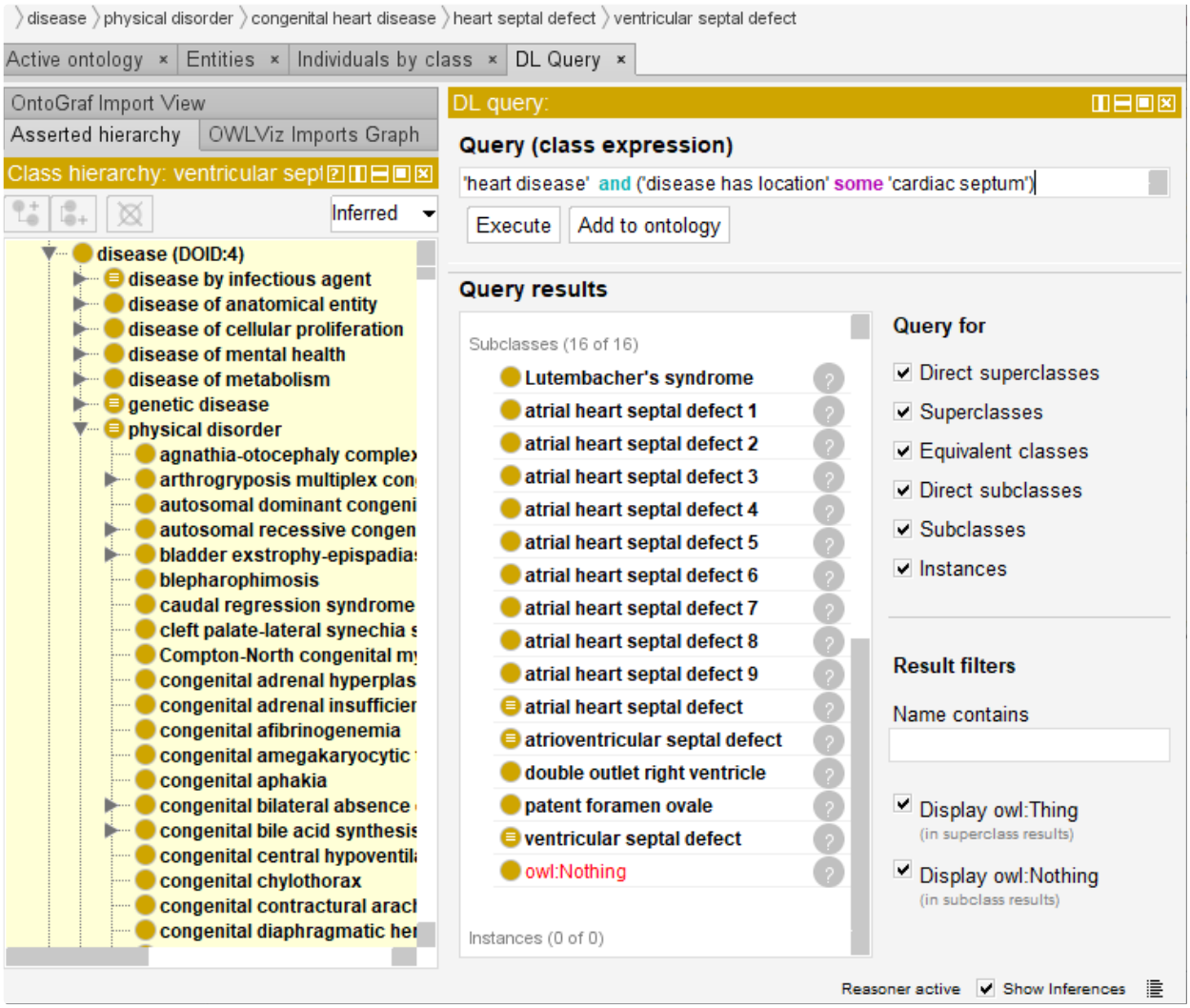


Figure 3.22: Continuation of inferred subclasses, from description logic query in Protégé (Figure 3.21).

Figure 3.22 presents a comprehensive overview of the subclasses of heart diseases, specifically those located in the cardiac septum. The subclasses, which encompass various specific types of atrial heart septal defects, atrioventricular septal defect, double outlet right ventricle, patent foramen ovale, and ventricular septal defect, are meticulously detailed under the class hierarchy. This comprehensive approach underscores the breadth and depth of conditions within the broader category of heart septal defects. However Figure 3.23a shows the direct subclasses of 'nutrient deficiency 'are:

- 'deficiency of biotin'
- 'deficiency of choline'
- 'deficiency of vitamin C'

A nutrient deficiency is the lack or insufficiency of essential nutrients necessary for normal body functioning. Such a classification system enhances the systematic organization of medical knowledge, facilitating better data retrieval, analysis, and clinical application. Additionally, understanding the different types of nutrient deficiencies and their relationships to disease drivers can aid in diagnosing and treating related conditions. Ontology is not just a static structure but can dynamically infer new relationships based on existing data. This is essential for applications in artificial intelligence and data analysis, where understanding implicit relationships is necessary.

$$\text{heartdisease} \sqcap \text{nutrient deficiency} \tag{12}$$

The query 12 in Figure 3.23b, illustrate that the super classes are:

- cardiovascular system disease
- disease driver
- disease of anatomical entity
- heart disease
- nutrient deficiency
- disease And direct super classes are:
- heart disease
- nutrient deficiency

Ontological inferences state that the disease is categorized as both a cardiovascular system disease and a disease of an anatomical entity.

Integrating the DOID Ontology with DL model offers a richer understanding by introducing concepts that may not be explicitly present in the training data. DOID includes a wide range of terms such as anatomy, disease drivers (environmental or genetic factors contributing to complex diseases), OMIM susceptibility, onset (the age group when a disease typically appears), phenotype, sequence, and transmission process (how a pathogen spreads from one source to another). This broader conceptual coverage enhances the model's ability to extract more precise and meaningful medical information.

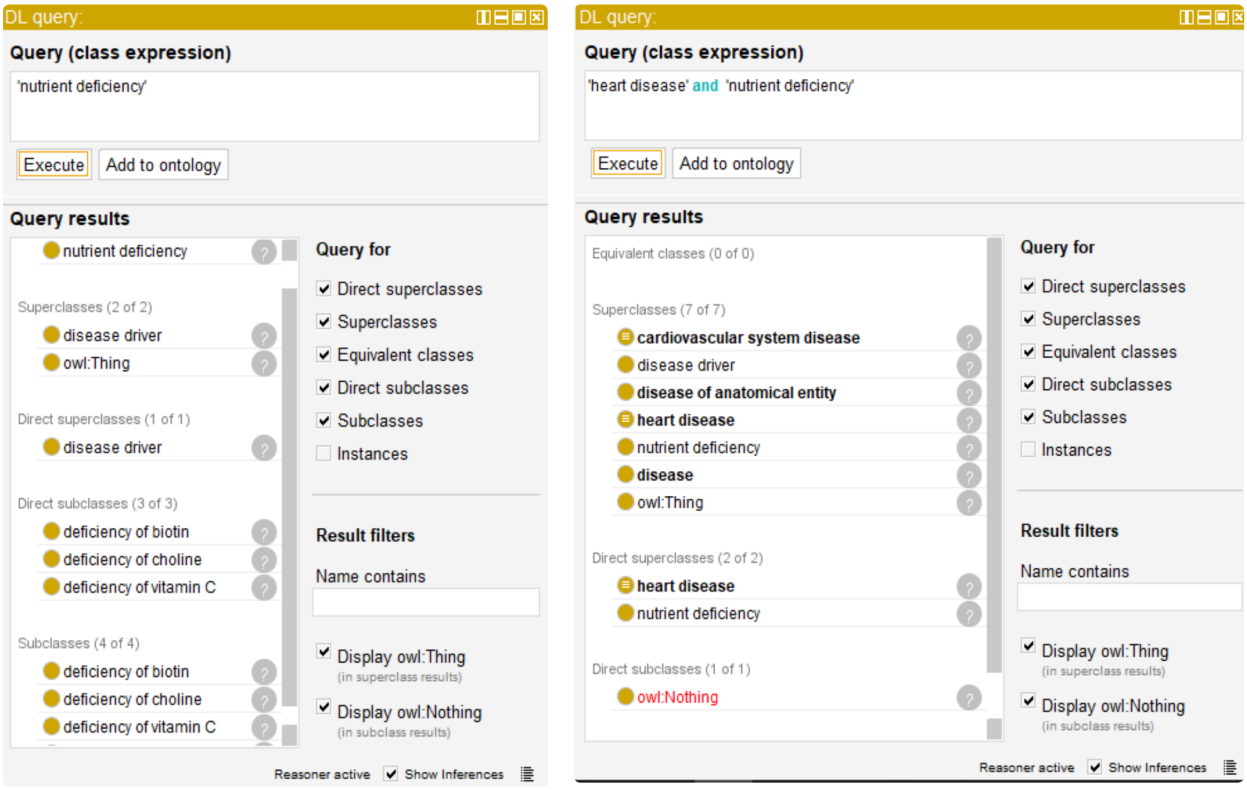


Figure 3.23: Inferred results of 'nutrient deficiency' and ('heart disease' and 'nutrient deficiency'), from Description logic query.

3.3.5 Jena API vs. Owlready2

This section presents a comparative evaluation of two prominent ontology management tools (Jena API and Owlready2) used in conjunction with the CONSEPTR DL architecture for extracting clinical events and temporal expressions (Table 3.11). On the i2b2-2012 dataset, both Jena API and Owlready2 exhibit high accuracy and F1-scores, with Owlready2 achieving slightly superior values across all ontology configurations. For instance, Owlready2 records an F1-score of 85.31 with SYMP, compared to Jena's 84.42. This consistency reflects the effectiveness of both tools in handling well-structured and annotated clinical narratives. Notably, the DRON, SYMP, and DOID ontologies combined with either tool yields the highest performance, demonstrating the benefit of integrating multiple domain-specific ontologies.

Performance on the i2b2-2010 dataset reflects a shift, where Jena API maintains an accuracy of 83% across all ontology settings with F1-scores ranging between 82.60 and 83.13. Owlready2, on the other hand, records slightly lower accuracy (80%) and F1-scores between

Table 3.11: Comparative summary of F1-Score and accuracy results for DL models and ontology-integrated approaches using Jena API and Owlready2 library.

Ontology's Tools	Datasets	CONSEPTR+DRON		CONSEPTR+SYMP		CONSEPTR+DOID		CONSEPTR+DSD	
		Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score
Jena Api	I2b2-2012	84	83.93	84	84.42	85	84.59	85	85.23
	I2b2-2010	83	82.60	83	83.13	83	82.89	83	82.60
	I2b2-2018	43	42.84	54	53.56	38	38.43	53	53.34
Owlready2	I2b2-2012	85	85.12	85	85.31	85	85.25	85	85.00
	I2b2-2010	80	79.97	80	79.51	80	79.58	81	80.98
	I2b2-2018	41	41.36	38	37.72	38	38.14	41	39.89

CONSEPTR					
I2b2-2012		I2b2-2010		I2b2-2018	
Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score
85	85.37	79	78.99	38	38.46

79.51 and 80.98. These results suggest that Jena API may offer better generalization capabilities.

The results on the i2b2-2018 dataset underscore the greater challenge posed by this corpus. Here, Jena API clearly outperforms Owlready2, achieving higher accuracy and F1-scores across all ontologies. For example, with the SYMP ontology, Jena achieves an F1-score of 53.56 compared to Owlready2's 37.72. This performance gap indicates that Jena API is more robust in processing unstructured and heterogeneous data, which is characteristic of the i2b2-2018 set.

A comparison with the baseline CONSEPTR model (applied without ontology integration) further highlights the value of semantic enrichment. While the standalone CONSEPTR model achieves an F1-score of 85.37 on i2b2-2012, its performance drops to 78.99 on i2b2-2010 and 38.46 on i2b2-2018. The integration of domain ontologies through either Jena or Owlready2 consistently improves upon these results, confirming that ontology-driven reasoning significantly enhances the extraction and classification of medical entities.

In summary, the comparative study reveals that while Owlready2 performs slightly better in structured contexts such as i2b2-2012, Jena API demonstrates greater adaptability and effectiveness in more complex environments like i2b2-2018. The overall findings support the conclusion that integrating ontologies with DL models not only boosts classification performance but also enriches semantic understanding in clinical NLP applications.

3.4 New Approach for temporal Relations Extraction

Medical professionals often require access to detailed and temporally organized patient information to effectively assess disease progression and evaluate treatment outcomes. To address this need, modern Electronic Medical Records (EMRs) frequently include clinical

narratives that capture temporal aspects of patient care. Despite the richness of this textual data, its full exploitation remains limited due to inherent linguistic complexity and technical challenges associated with NLP in the clinical domain.

Let i denote a temporal interval within the set I, where $i = [t_1, t_2]$, and $t_1, t_2 \in T$ such that $t_1 < t_2$. Given T be the set of measured time values, structurally equivalent to the real number line $\mathbb{R}$, and I represents the set of defined time intervals, temporal relationships between two such intervals are typically formalized using Allen's interval algebra [186], which defines a set of fundamental temporal operators. These operators describe possible relations between intervals, including: *before*, *meets*, *overlaps*, *starts*, *during*, *finishes*, and *equals*, along with their respective inverses: *after*, *met-by*, *overlapped-by*, *started-by*, *contains*, and *finished-by*. Notably, the *equals* operator is symmetric.

As illustrated in Figure 3.24, each operator is defined in terms of the start and end points of intervals, denoted as Begin(i) and End(i). These relations form the foundation for temporal reasoning in various artificial intelligence applications. While Allen initially introduced these operators in 1984, subsequent work reaffirmed their utility.

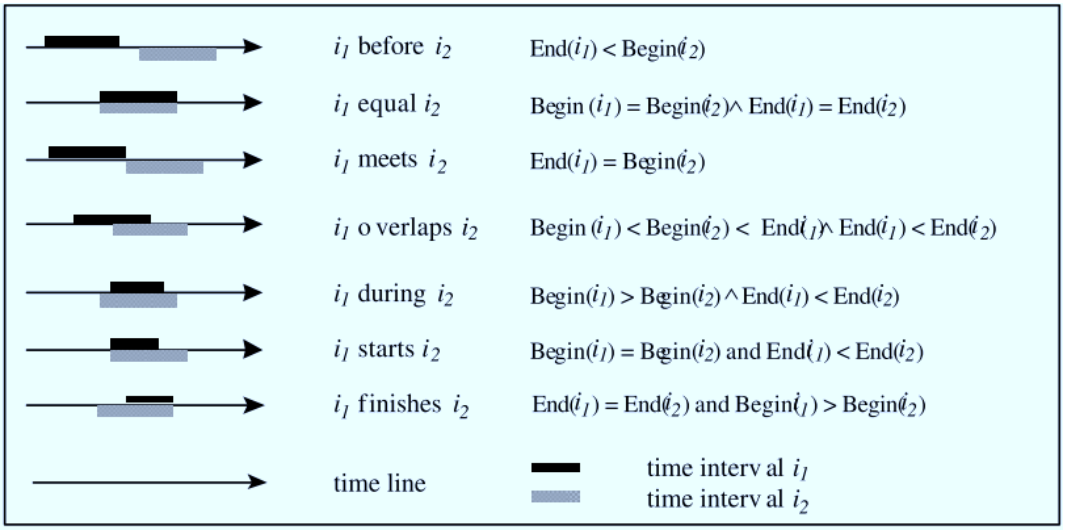


Figure 3.24: Fundamental temporal relationships between two time intervals as defined by Allen's interval algebra [187].

3.4.1 Dataset and Pre-Processing Process

The i2b2-2012 dataset, as detailed in paragraph 3.2, is utilized to extract temporal link. To prepare the temporal relation (Tlink) data for DL tasks, a structured preprocessing algorithm was designed (5). Data information is first extracted from XML files and consolidated into a single text file for the test dataset. To ensure consistency and standardization, all extracted

textual information is converted to uppercase. Next, sequences corresponding to temporal links (Tlinks) are identified and extracted based on their start and end positions within the text. The training and test datasets are then merged to form a unified dataset, which is subsequently split using a balanced sampling strategy to preserve class distribution. This results in two subsets: `train_link`, comprising 75% of the data, and `test_link`, containing the remaining 25%. Post-splitting, both subsets undergo a cleaning process where duplicate sequences are removed, null sequences are discarded, and sequences with missing relation labels are filtered out. This ensures that the final datasets used for model training and evaluation are both representative and of high quality. Figure 3.25b illustrates that the

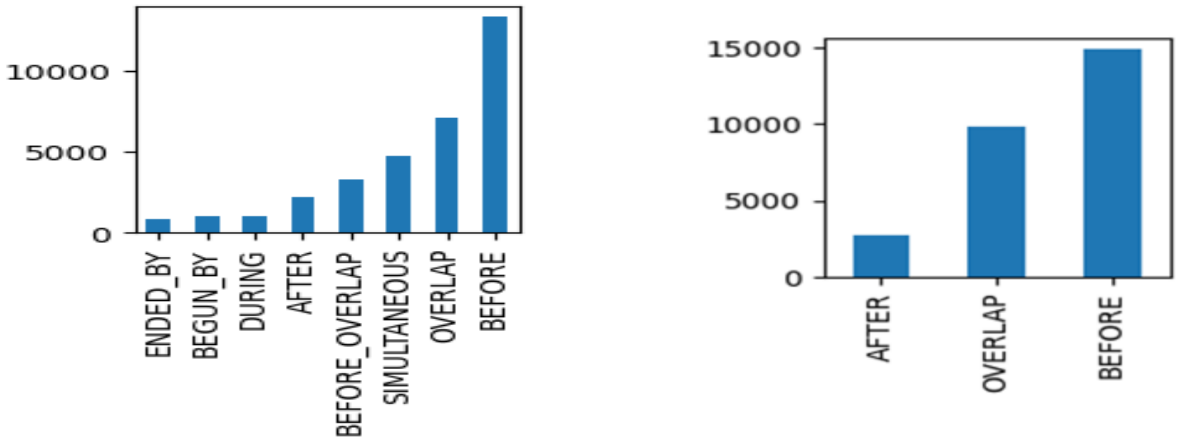
Algorithm 5 Data preparation for TLink

Require: i2b2 datasets (train and test)
Ensure: `Train dataset (train_link)`, `Test dataset (test_link)`
1: Extract data information from XML files into a single text files for train and test data
2: Extract the sequences of the Tlink using `(start, end)` information
3: Convert all extracted information to uppercase
4: Merge training and test datasets into `Data`
5: Split `Data` using balanced sampling:
6: `train_link` ← 75% of `Data`
7: `test_link` ← 25% of `Data`
8: Let $R \leftarrow \{$`train_link`, `test_link`$\}$
9: **for all** $e \in R$ **do**
10: Delete duplicate sequences from e
11: Delete sequences with null values from e
12: Delete sequences with null labels for relations from e
13: **end for**

test corpus contains only a limited subset of temporal relation types—specifically, AFTER, OVERLAP, and BEFORE. As a result, it is not feasible to evaluate model performance on the remaining relation types that are absent from the test dataset. To address this limitation, our data preparation strategy, detailed in Algorithm 5, ensures that all relation labels present in the training dataset are also represented in the test dataset. This balanced distribution, shown in Figure 3.26, enables more comprehensive training and robust evaluation of the learning model.

3.4.2 Models Description

To encode the semantic content of candidate sequences, we utilized the `nnlm-en-dim50/2` sentence embedding model, which is based on feed-forward Neural Network Language Models (NNLMs) [188]. This embedding model offers 50-dimensional pre-trained representations and



(a) Train data distribution.

(b) Test data distribution.

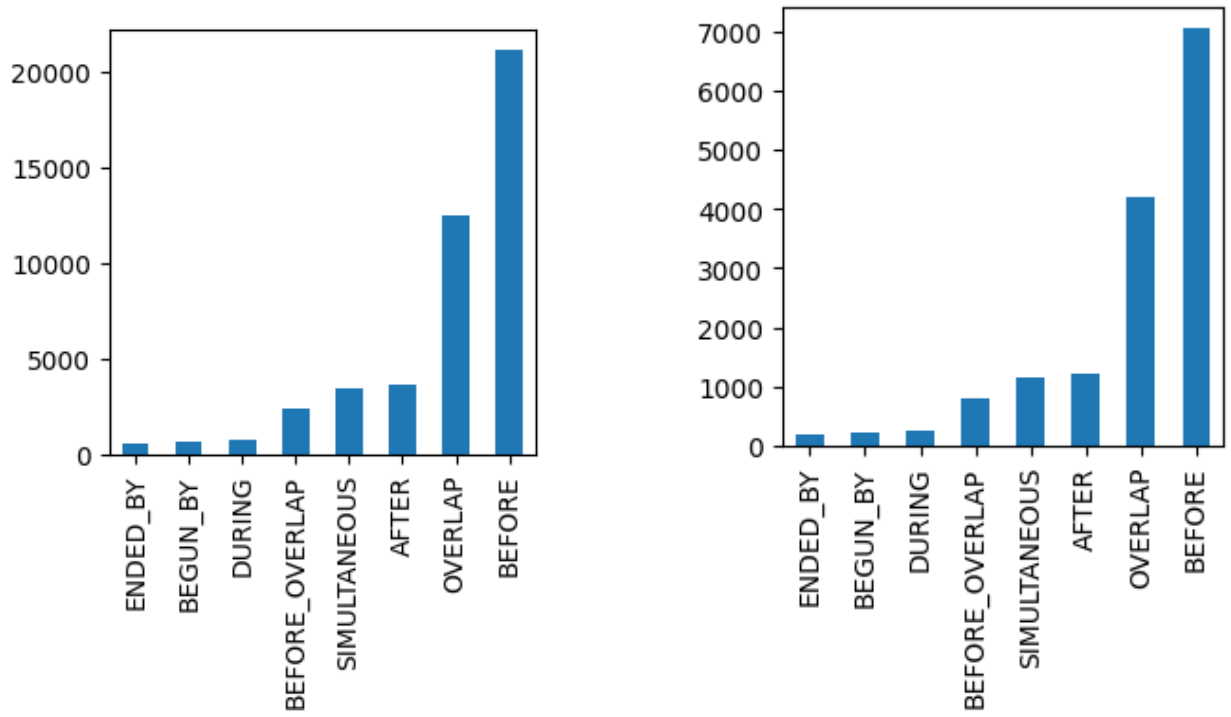
Figure 3.25: TLink distribution in train and test datasets before merging train and test datasets (original datasets).

effectively manages out-of-vocabulary (OOV) terms.

In order to classify the temporal relations types in clinical narratives, we investigated the performance of four convolutional neural network (CNN) architectures in our experiments:

1. Standard convolutional layer,
2. Transposed convolutional layer model,
3. Separable 1D convolutional model, and
4. ConvLSTM 1D model, which integrates convolutional operations with temporal recurrence mechanisms.

In the proposed architecture, the sentences are numerically encoded using the NNLM embedding model, loaded through a `hub.KerasLayer`. The convolutional layers utilize 128 filters with a kernel size of 1, and the activation function is set to `ReLU`. A flattening layer transforms the convolutional outputs into a one-dimensional vector, which is then passed to a Dense layer. This Dense layer contains eight neurons corresponding to the eight relation types, or three neurons, with a sigmoid activation function applied.



(a) Train data distribution.

(b) Test data distribution.

Figure 3.26: TLink distribution in training and testing datasets after merging train and test datasets.

3.4.3 Results and discussion

Table 3.12 summarizes the performance comparison among several convolution-based architectures employed for temporal relation extraction, including ConvLSTM, Standard Convolution, Separable Convolution, and Transposed Convolution. The evaluation is conducted under three distinct experimental configurations:

1. Utilizing the complete set of temporal relation labels.
2. Restricting the test dataset to the relation types: AFTER, OVERLAP, and BEFORE.
3. Limiting both the training and test datasets to the relation types: AFTER, OVERLAP, and BEFORE.

Table 3.12 provides a comprehensive comparative evaluation of four convolution-based architectures (LSTM, Standard Convolution, Separable Convolution, and Transposed Convolution) under three distinct experimental configurations aimed at extracting temporal relations from clinical narratives.

In the first setting, which includes all eight defined temporal relation types in both training and testing phases, convolutional models significantly outperformed LSTM during

Table 3.12: Global performance results for TLink models on the training and test datasets.

Used Data	Subset (Train/Test)	Metric	LSTM	Standard	Separable	Transposed
Using dataset for all defined relation type (all data sets)	Train	Accuracy	82.34	95.51	95.35	95.39
		Precision	53.37	66.46	59.71	61.28
		Recall	91.59	98.65	98.61	98.68
	Test	Accuracy	66.69	67.44	67.52	67.69
		Precision	46.16	49.53	45.37	46.16
		Recall	79.48	76.65	77.55	77.70
Test datasets with AFTER, OVERLAP, and BEFORE	Train	Accuracy	55.27	65.24	65.72	65.79
		Precision	41.03	43.18	49.00	49.00
		Recall	69.56	83.87	79.69	79.94
	Test	Accuracy	73.27	59.17	59.44	58.52
		Precision	47.14	37.11	44.47	43.13
		Recall	78.96	74.89	68.85	70.67
Train and test datasets with AFTER, OVERLAP, and BEFORE	Train	Accuracy	83.05	97.40	97.01	97.23
		Precision	79.11	93.93	93.40	93.02
		Recall	85.65	98.15	97.53	98.26
	Test	Accuracy	79.56	80.51	80.82	80.55
		Precision	75.76	77.29	77.29	76.59
		Recall	82.34	82.61	82.85	82.90

training, with accuracies exceeding 95%. However, this high training performance did not translate into comparable generalization on the test dataset, where accuracy across all models converged around 67%. The considerable gap between training and test accuracy suggests a degree of overfitting, possibly due to increased classification complexity from the full relation set. Despite lower precision, the LSTM model demonstrated relatively high recall, indicating sensitivity to detecting temporal relations but with a higher false positive rate.

In the second configuration, where the test dataset was limited to three major relation types—AFTER, OVERLAP, and BEFORE—while training remained on the full relation set, the LSTM model achieved the highest test accuracy (73.27%), outperforming all convolutional models. This suggests that LSTM is better at generalizing to simplified label distributions not seen during training. In contrast, convolutional models exhibited decreased performance, likely due to mismatches between training and evaluation label sets, emphasizing the importance of distributional alignment.

The third configuration restricted both training and testing to only the three major relation types, resulting in markedly improved performance for all models. Convolution-based models, particularly Standard and Separable Convolutions, achieved over 97% accuracy on the training set and approximately 80–81% on the test dataset, with balanced precision and recall values. These results highlight the suitability of CNN-based architectures for temporal relation extraction in low-variance label spaces.

Based on these findings (Tables 3.12 and 3.13), we propose two ensemble models to leverage the strengths of individual classifiers through majority voting. The first, denoted

Table 3.13: Detailed results of relation extraction, displaying the performance for each relation type.

Used Data	Relation Type	LSTM			Standard			Separable			Transposed		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Using dataset for all defined relation type (all data sets)	ENDED_BY	0.28	0.11	0.16	0.18	0.15	0.17	0.21	0.19	0.20	0.20	0.15	0.17
	BEGUN_BY	0.26	0.26	0.26	0.27	0.23	0.25	0.23	0.20	0.21	0.30	0.19	0.23
	DURING	0.38	0.31	0.34	0.38	0.28	0.32	0.37	0.25	0.30	0.34	0.31	0.32
	BEFORE_OVERLAP	0.33	0.34	0.33	0.35	0.33	0.34	0.37	0.30	0.33	0.39	0.28	0.33
	SIMULTANEOUS	0.46	0.41	0.43	0.44	0.43	0.44	0.42	0.42	0.42	0.42	0.47	0.44
	AFTER	0.37	0.26	0.30	0.39	0.30	0.34	0.39	0.31	0.35	0.41	0.33	0.36
	OVERLAP	0.60	0.67	0.63	0.63	0.63	0.63	0.62	0.66	0.64	0.62	0.65	0.64
	BEFORE	0.84	0.86	0.85	0.83	0.89	0.86	0.85	0.88	0.86	0.84	0.88	0.86
Test datasets with AFTER, OVERLAP, and BEFORE	AFTER	0.00	0.00	0.00	0.36	0.04	0.07	0.27	0.10	0.14	0.25	0.08	0.12
	OVERLAP	0.63	0.81	0.71	0.65	0.43	0.52	0.65	0.43	0.52	0.64	0.43	0.51
	BEFORE	0.81	0.82	0.82	0.84	0.80	0.82	0.84	0.79	0.81	0.85	0.78	0.81
Train and Test datasets with AFTER, OVERLAP, and BEFORE	AFTER	0.48	0.28	0.35	0.51	0.35	0.41	0.53	0.33	0.41	0.49	0.38	0.43
	OVERLAP	0.72	0.82	0.76	0.75	0.79	0.77	0.76	0.79	0.78	0.75	0.79	0.77
	BEFORE	0.88	0.87	0.88	0.87	0.89	0.88	0.86	0.90	0.88	0.88	0.89	0.88

as R_CONSEPTRL_8, aggregates predictions from all four models trained under the full eight-relation configuration (first experimental setup). This ensemble aims to mitigate overfitting and improve generalization by combining diverse inductive biases. The second model, R_CONSEPTRL_3, fuses outputs from models trained under the aligned three-relation setting (third configuration), capitalizing on the observed performance consistency in that scenario. These ensemble approaches are expected to enhance both accuracy and robustness in temporal relation classification, offering a practical solution for real-world clinical applications with variable annotation complexity and label sparsity. Table 3.14 reports the test per-

Table 3.14: Test results for the models R_CONSEPTRL_8 and R_CONSEPTRL_3.

Performances test results	F1-score	Accuracy
R_CONSEPTRL_8	69.20	69
R_CONSEPTRL_3	81.33	81

formance of the ensemble models R_CONSEPTRL_8 and R_CONSEPTRL_3, evaluated using F1-score and overall accuracy. These ensemble models were constructed via majority voting across individual classifiers, with R_CONSEPTRL_8 aggregating predictions from models trained on the complete set of eight temporal relation types, and R_CONSEPTRL_3 utilizing models trained exclusively on the three most frequent relations: AFTER, OVERLAP, and BEFORE.

The results demonstrate a clear performance distinction between the two ensemble strategies. R_CONSEPTRL_3 achieved an accuracy of 81% and a F1-score of 81.33%, substantially outperforming R_CONSEPTRL_8, which recorded 69% accuracy and 69.20% F1-score. This performance gap highlights the impact of label set complexity on model generalization. Specifically, the reduced relation space in R_CONSEPTRL_3 likely contributed to

more reliable predictions due to lower class imbalance and fewer decision boundaries, which also aligns with findings from earlier experiments.

Furthermore, the superior performance of R_CONSEPTRL_3 suggests that targeted training on high-frequency relation subsets can lead to more robust and generalizable models in real-world clinical scenarios, where noisy or infrequent relation types may introduce ambiguity. In contrast, the broader scope of R_CONSEPTRL_8, although more comprehensive, appears to introduce increased classification difficulty, reflected in the lower precision and recall values observed in earlier analyses.

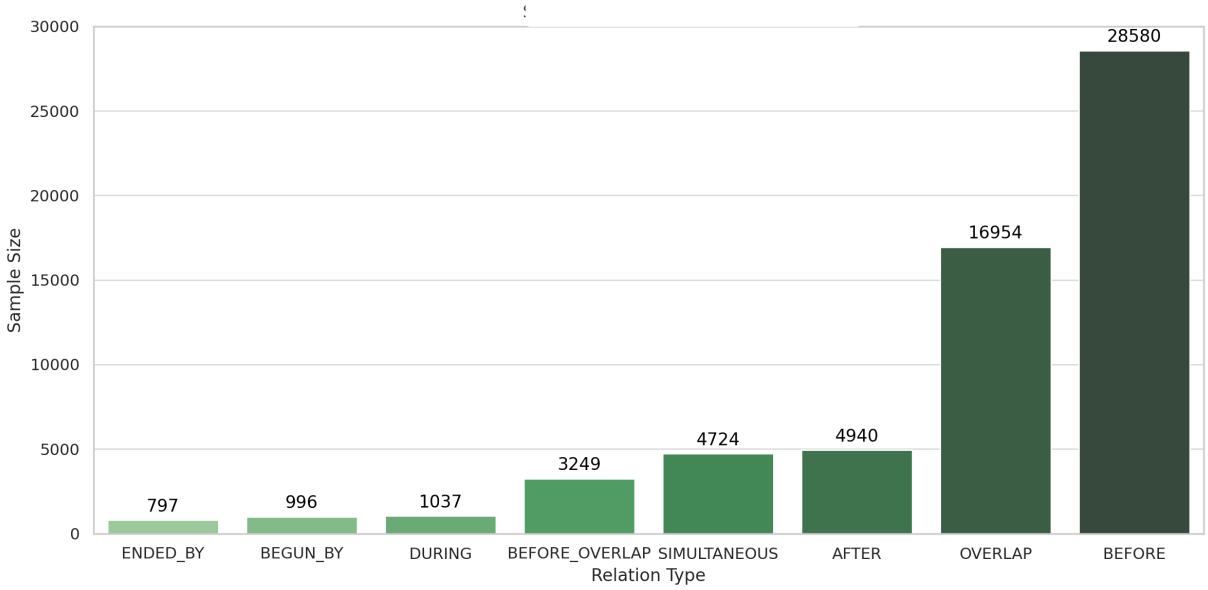


Figure 3.27: Sample size by relation type for R_CONSEPTRL_8 model.

The relationship between sample size and F1-score for the eight relation types(as presented in Figures 3.27 and 3.28) reveals a clear trend: larger sample sizes generally correspond to higher F1-scores, indicating better model performance. For example, the relation type BEFORE, with the largest sample size of 28,580 instances, achieves the highest F1-score of 0.87. Similarly, OVERLAP, with 16,954 samples, attains a strong F1-score of 0.66. In contrast, smaller datasets such as ENDED_BY, which contains only 797 samples, correspond to a low F1-score of 0.18. This suggests that having a more substantial amount of data significantly enhances the ability of the model to learn and generalize effectively for the given relation type. Moreover, some variability in F1-scores among relation types with similar sample sizes indicates that sample size is not the only determinant of model accuracy. For instance, both BEFORE_OVERLAP and AFTER have comparable sample sizes (3,249 and 4,940 respectively), but their F1-scores differ, with BEFORE_OVERLAP achieving 0.36

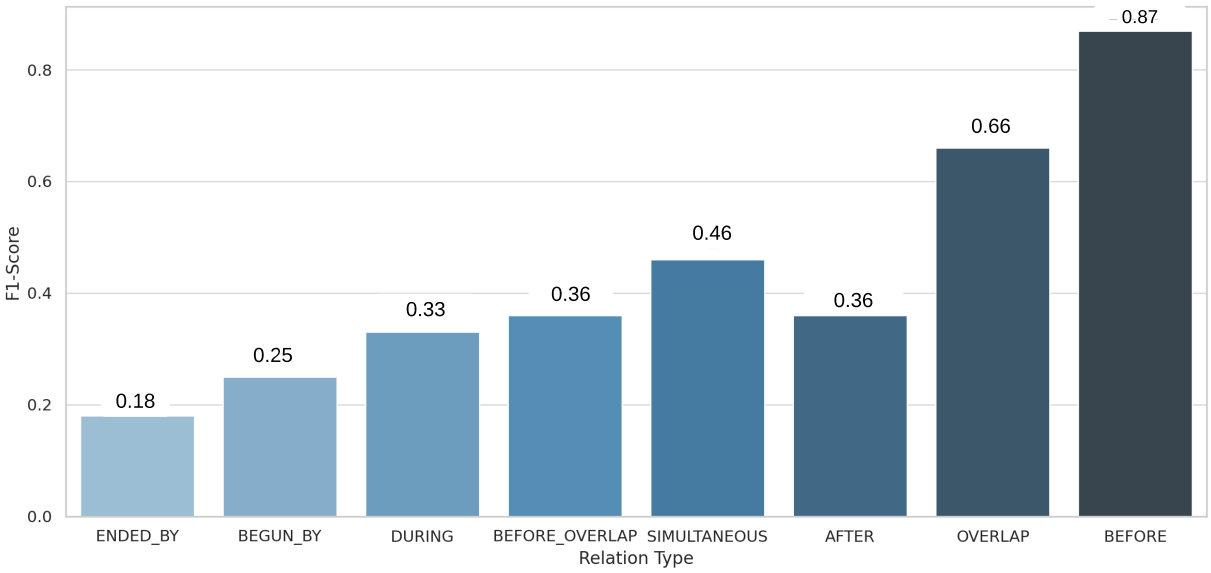


Figure 3.28: F1-Score by relation type for R_CONSEPTRL_8 model.

and SIMULTANEOUS, with a similar size (4,724 samples), reaching 0.46. This variability likely reflects differences in the complexity of the relation classes, the quality of annotations, or the inherent difficulty of distinguishing those relations, suggesting that model performance depends on multiple interacting factors.

In conclusion, while sample size plays a critical role in determining F1-score performance in relation classification tasks, it is not the sole factor. Understanding and addressing other challenges—such as class complexity and annotation quality—alongside increasing data quantity will be essential for achieving further performance improvements. Figure 3.29 presents a comparison of F1-scores across recent studies using the i2b2-2012 dataset clearly highlights the competitive performance of the proposed models, R_CONSEPTRL_3 and R_CONSEPTRL_8, relative to prior state-of-the-art approaches. This analysis evaluates the models in terms of both relation type coverage and overall accuracy, as measured by the F1-score metric. Among the evaluated models, the R_CONSEPTRL_3 model achieves an F1-score of 81.33, surpassing all but one of the previously published approaches. Notably, it outperforms high-ranking models such as those reported in Alfattni et al. (2021) [87] (81.10), Zhou et al. (2020) [86] (80.20), and [76] (67.10), affirming its robustness when handling standard temporal relations like AFTER, BEFORE, and OVERLAP (i.e., TLink-a types). Its superior performance indicates that the architectural enhancements and learning strategies introduced in R_CONSEPTRL_3 have led to more effective temporal relation classification. Similarly, the R_CONSEPTRL_8 model achieves a notable F1-score

of 69.20, which is competitive with and in several cases superior to many existing models targeting the broader and more complex set of Allen interval algebra relations (TLink-b). For instance, R_CONSEPTRL_8 outperforms models such as Grouin et al. (2013) [83] (62.31), [84] (53.70), and [189] (34.10), and is only marginally below the score reported by Haq et al. (2021) [88] (73.60). Given the inherent difficulty of modeling nuanced temporal relations like BEGUN_BY, DURING, and SIMULTANEOUS, this performance reflects a promising advancement in temporal reasoning capabilities. In addition to these comparisons, other approaches represented in the figure include models proposed by Chang et al. (2013) [190], Cherry et al. (2013) [191], Lin et al. (2016) [192], Tang et al. (2013) [93], D'Souza and Ng (2013) [193], Nikfarjam et al. (2015) [194], Cheng et al. (2013) [195], Lee et al.(2018) [196], Saiz and Altuna (2023) [94], and Cheng and Weiss (2023) [197]. While many of these models report reasonable performance within specific temporal frameworks, the proposed R_CONSEPTRL models consistently demonstrate superior or comparable effectiveness across both TLink-a and TLink-b categories. Overall, both proposed models demonstrate a strong generalization capability across different temporal relation schemes. Their F1-scores are consistently above the majority of existing benchmarks, indicating both methodological soundness and practical applicability. The high performance of R_CONSEPTRL_3, in particular, positions it among the top-performing approaches to date for temporal relation extraction on the i2b2-2012 dataset. This reinforces the potential impact of the proposed methodology on advancing temporal understanding in clinical narratives and supports its adoption for downstream temporal information extraction tasks in medical NLP.

3.4.4 Temporal Link (TLink) Prediction Approach

To formalize the extraction of temporal relationships, we define a classification function F as follows:

$$F(s_{ij}) = y', \quad \text{where } y' \in R, \quad R = \left\{ \begin{array}{l} \text{ENDED_BY, BEGUN_BY, DURING,} \\ \text{BEFORE_OVERLAP, SIMULTANEOUS,} \\ \text{AFTER, OVERLAP, BEFORE} \end{array} \right\}$$

In this formulation, s_{ij} denotes a candidate sequence representing the contextual span between concept i and concept j. The function F maps this sequence to a specific temporal relation label y' selected from the predefined set R of possible temporal relation types.

The classification model is designed to evaluate all possible concept pairs in the data,

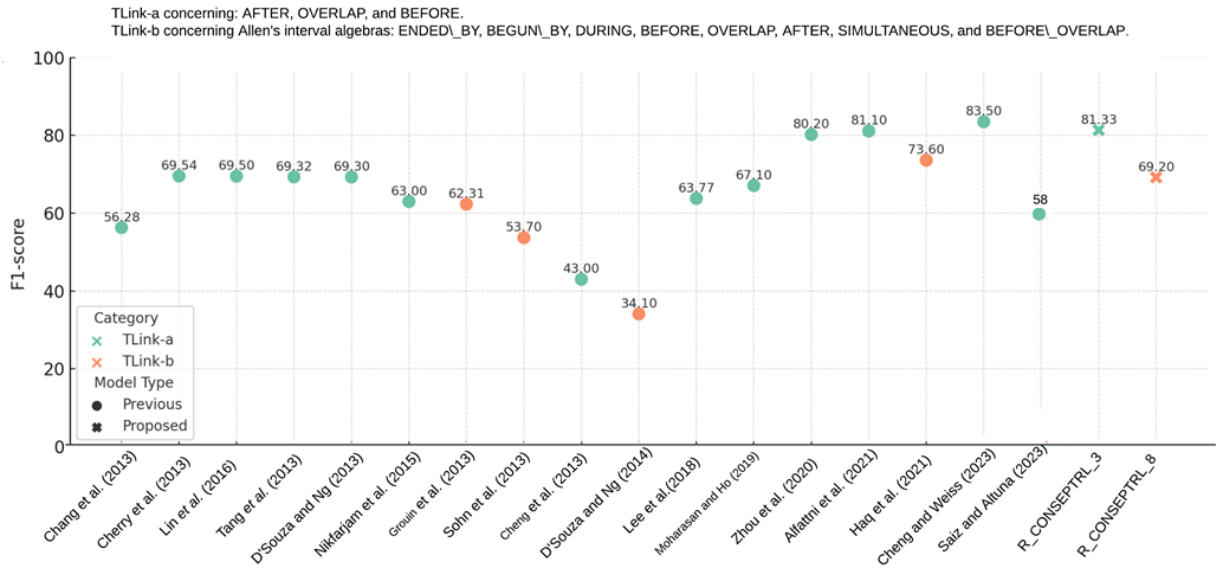


Figure 3.29: Performance comparison with studies used i2b2-2012 datasets.

with the exception of those where both concepts belong to the TIMEX category. Formally, for all $x_i \in E$, and $x_j \in E$, where

$$E = \{Event \cup TIMEX\}$$

and

$$(x_i, x_j) \notin \text{TIMEX} \times \text{TIMEX},$$

there exists a relation $y' \in R$ such that $F(s_{ij}) = y'$, and s_{ij} denotes the corresponding candidate sequence.

To extract the temporal relations `ENDED_BY`, `BEGUN_BY`, `DURING`, `BEFORE_OVERLAP`, `AFTER`, `SIMULTANEOUS`, `OVERLAP`, and `BEFORE`, four convolutional neural network architectures were employed in our experiments: Standard Convolutional, Separable Convolutional, Transposed Convolutional, and LSTM Convolutional. This ensemble is referred to as `R_CONSEPTRL_8` (Relational model combining Standard, Transposed, Separable, and LSTM Convolutional layers for all eight defined relation types).

Prior work primarily focused on extracting a subset of temporal relations, namely `AFTER`, `OVERLAP`, and `BEFORE`. Accordingly, we conducted additional experiments evaluating the performance of the model `R_CONSEPTRL_3`, which utilizes Standard, Transposed, and Separable Convolutional layers to classify these three relation types. In this approach, temporal relation extraction relies on a majority voting scheme among the model outputs.

Algorithm 6 Temporal relation extraction using R_CONSEPTRL_8 Model

Require: Clinical text

Ensure: Annotated text with temporal relations in XML format

Extract events and temporal expressions from input text using CONSEPTR model and Jena API ([198]); Result: *Events and TIMEX List*

Generate candidate sentence pairs based on extracted entities: List

for each candidate sequence in List **do**

Pass candidate sequence through four CNN models:

- Standard Convolution Model → Output1
- Transposed Convolution Model → Output2
- Separable Convolution Model → Output3
- LSTM Convolution Model → Output4

Model outputs = $\{Output_1, Output_2, Output_3, Output_4\}$,

confidence threshold $\theta = 0.70$

Predictions list $P \leftarrow \emptyset$

for each model output $Output_i$ **do**

Compute class index $c_i = \arg\max(Output_i)$

Compute confidence score $s_i = \max(Output_i)$

if $s_i \geq \theta$ **then**

Append c_i to P

end if

end for

if $P = \emptyset$ **then**

Exclude candidate sentence from consideration

else

Final prediction $\hat{y} \leftarrow \text{MajorityVote}(P)$

end if

end for

Annotate medical text

Store annotated output in XML format for downstream use with clinical data

The modular design of the proposed model reduces complexity and development costs by enabling reusable components that can be dynamically combined. The overall methodology for temporal relation prediction is illustrated in Algorithm 6, which demonstrates the integration of the CONSEPTR model with the Jena API as presented in [198] for event and temporal expression prediction. Subsequently, candidate sequences are extracted for

relation classification. Additionally, the temporal relation extraction process considers only predictions with confidence scores greater than or equal to 70%, representing the majority consensus across the four models.

Example Prediction: To demonstrate the effectiveness of the proposed system in extracting and classifying temporal information from clinical narratives, we present an example using a de-identified patient record processed through our integrated framework, which includes CONSEPTR with Jena API for event and temporal expression extraction and R_CONSEPTRL_8 for temporal relation classification. The semantic enrichment is performed using the Jena API, enabling structured representation and reasoning over the extracted data.

The input clinical narrative describes a 47-year-old female patient with type 1 diabetes mellitus, who underwent a postcadaveric kidney transplantation and currently has good graft function. Complications of the disease include retinopathy, nephropathy, and peripheral neuropathy. A temporal phrase, "four months", is also identified within the context of the transplantation event. The following key events were automatically extracted and

```
<?xml version="1.0" encoding="UTF-8"?>
<ClinicalNarrativeTemporalAnnotation>
 <TEXT>
<![CDATA[
 The patient is a 47 year old woman with type 1 diabetes mellitus, who took postcadaveric kidney transplantation for four months and now has good graft function.
 Her diabetes mellitus has been complicated by retinopathy and nephropathy in common with peripheral neuropathy.
 ]]></TEXT>
 <TAGS>
 <EVENT id="E0" start="41" end="65" text="TYPE 1 DIABETES MELLITUS" type="Disease" />
 <EVENT id="E1" start="76" end="112" text="POSTCADAVERIC KIDNEY TRANSPLANTATION" type="TREATMENT" />
 <EVENT id="E2" start="141" end="160" text="GOOD GRAFT FUNCTION" type="OCCURRENCE" />
 <EVENT id="E3" start="208" end="219" text="RETINOPATHY" type="PROBLEM" />
 <EVENT id="E4" start="224" end="235" text="NEPHROPATHY" type="PROBLEM" />
 <EVENT id="E5" start="251" end="272" text="PERIPHERAL NEUROPATHY" type="PROBLEM" />
 <TIMEX id="T0" start="117" end="128" text="FOUR MONTHS" type="DURATION" />
 <TLINK id="TL0" fromID="E1" fromText="POSTCADAVERIC KIDNEY TRANSPLANTATION" toID="E0" toText="TYPE 1 DIABETES MELLITUS" type="OVERLAP" />
 <TLINK id="TL1" fromID="T0" fromText="FOUR MONTHS" toID="E0" toText="TYPE 1 DIABETES MELLITUS" type="OVERLAP" />
 <TLINK id="TL2" fromID="E2" fromText="GOOD GRAFT FUNCTION" toID="E0" toText="TYPE 1 DIABETES MELLITUS" type="OVERLAP" />
 <TLINK id="TL3" fromID="E3" fromText="RETINOPATHY" toID="E0" toText="TYPE 1 DIABETES MELLITUS" type="OVERLAP" />
 <TLINK id="TL4" fromID="E4" fromText="NEPHROPATHY" toID="E0" toText="TYPE 1 DIABETES MELLITUS" type="OVERLAP" />
 <TLINK id="TL5" fromID="E5" fromText="PERIPHERAL NEUROPATHY" toID="E0" toText="TYPE 1 DIABETES MELLITUS" type="OVERLAP" />
 <TLINK id="TL6" fromID="T0" fromText="FOUR MONTHS" toID="E1" toText="POSTCADAVERIC KIDNEY TRANSPLANTATION" type="SIMULTANEOUS" />
 <TLINK id="TL7" fromID="E2" fromText="GOOD GRAFT FUNCTION" toID="E1" toText="POSTCADAVERIC KIDNEY TRANSPLANTATION" type="OVERLAP" />
 <TLINK id="TL8" fromID="E3" fromText="RETINOPATHY" toID="E1" toText="POSTCADAVERIC KIDNEY TRANSPLANTATION" type="OVERLAP" />
 <TLINK id="TL9" fromID="E4" fromText="NEPHROPATHY" toID="E1" toText="POSTCADAVERIC KIDNEY TRANSPLANTATION" type="OVERLAP" />
 <TLINK id="TL10" fromID="E5" fromText="PERIPHERAL NEUROPATHY" toID="E1" toText="POSTCADAVERIC KIDNEY TRANSPLANTATION" type="OVERLAP" />
 <TLINK id="TL11" fromID="E2" fromText="GOOD GRAFT FUNCTION" toID="T0" toText="FOUR MONTHS" type="BEFORE" />
 <TLINK id="TL12" fromID="E3" fromText="RETINOPATHY" toID="T0" toText="FOUR MONTHS" type="BEFORE" />
 <TLINK id="TL13" fromID="E4" fromText="NEPHROPATHY" toID="T0" toText="FOUR MONTHS" type="OVERLAP" />
 <TLINK id="TL14" fromID="E5" fromText="PERIPHERAL NEUROPATHY" toID="T0" toText="FOUR MONTHS" type="OVERLAP" />
 <TLINK id="TL15" fromID="E3" fromText="RETINOPATHY" toID="E2" toText="GOOD GRAFT FUNCTION" type="BEFORE\_OVERLAP" />
 <TLINK id="TL16" fromID="E4" fromText="NEPHROPATHY" toID="E2" toText="GOOD GRAFT FUNCTION" type="OVERLAP" />
 <TLINK id="TL17" fromID="E5" fromText="PERIPHERAL NEUROPATHY" toID="E2" toText="GOOD GRAFT FUNCTION" type="OVERLAP" />
 <TLINK id="TL18" fromID="E4" fromText="NEPHROPATHY" toID="E3" toText="RETINOPATHY" type="OVERLAP" />
 <TLINK id="TL19" fromID="E5" fromText="PERIPHERAL NEUROPATHY" toID="E3" toText="RETINOPATHY" type="OVERLAP" />
 <TLINK id="TL20" fromID="E5" fromText="PERIPHERAL NEUROPATHY" toID="E4" toText="NEPHROPATHY" type="OVERLAP" />
 </TAGS>
</ClinicalNarrativeTemporalAnnotation>
```

Figure 3.30: XML file displaying detailed annotations of a patient's events, temporal expressions, and temporal relation.

semantically categorized as presented in Figure 3.30:

- E0: Type 1 diabetes mellitus – Disease
- E1: Postcadaveric kidney transplantation – Treatment
- E2: Good graft function – Occurrence
- E3: Retinopathy – Problem
- E4: Nephropathy – Problem
- E5: Peripheral neuropathy – Problem
- The system also identifies a temporal expression:
- T0: Four months – Duration

The R_CONSEPTRL_8 model is then applied to determine temporal relationships (TLinks) between these events and temporal expressions. The resulting temporal links include both simple and composite relations from Allen's interval algebra. Key examples include:

- OVERLAP(E1, E0): The transplantation occurred during the course of diabetes.
- SIMULTANEOUS(T0, E1): The transplantation and the duration of four months coincide.
- OVERLAP(E2, E1): The graft function continues alongside the transplantation.
- BEFORE(E2, T0): Graft function started before the referenced four-month interval.
- BEFORE_OVERLAP(E3, E2): Retinopathy preceded and partially overlapped with the onset of graft function.
- OVERLAP(E4, E3), OVERLAP(E5, E3): Complications such as nephropathy and peripheral neuropathy occur alongside each other and with retinopathy.

These results exemplify the system's capacity to resolve fine-grained temporal relationships even in clinically complex and temporally ambiguous contexts. The use of CONSEPTR with Jena API enables rich semantic representation and supports temporal reasoning through ontologically informed linking of events and time expressions. Meanwhile, the robust architecture of R_CONSEPTRL_8 allows it to capture nuanced relationships including BEFORE_OVERLAP and SIMULTANEOUS, which are critical for constructing accurate patient timelines.

Overall, this example reinforces the practical utility and accuracy of our approach in clinical temporal information extraction. It highlights the system's ability to support downstream applications such as patient trajectory modeling, disease progression analysis, and decision support systems in medical informatics.

3.5 Conclusion

This chapter has presented a comprehensive and structured exploration of temporal information extraction from clinical narratives, integrating advanced DL techniques with ontology-based semantic reasoning. Through a multi-model methodology, we demonstrated how combining neural architectures with structured knowledge can significantly improve the accuracy, depth, and interpretability of medical text analysis. The first part of the chapter highlighted the effectiveness of DL models, particularly the CONSEPTR architecture, in extracting clinical events and TIMEX from unstructured text. Building upon this, the second component introduced semantic technologies to enhance the representational depth and inference capabilities of the system. The Jena API and Owlready2 frameworks were employed to manage ontologies, perform logical reasoning, and support knowledge-driven annotation workflows. Their comparative evaluation provided key insights into the practical benefits. The final section focused on temporal relation extraction using the proposed R_CONSEPTRL_8 model, which incorporate standard, separable, and transposed convolutional and ConvLSTM models within a majority voting ensemble. These models not only achieved high F1-scores across Allen's temporal relation types, but also demonstrated robust generalization. An illustrative prediction example further confirmed the model's capability to infer nuanced relationships between events and time expressions within a clinical context. The findings from this chapter affirm that the integration of DL with ontology-based reasoning substantially enhances temporal information extraction from clinical narratives. The proposed methodologies offer scalable, accurate, and semantically enriched solutions that can be extended to broader medical NLP tasks. This work lays the groundwork for future innovations in clinical decision support, patient timeline construction, and knowledge-based medical inference systems.

Chapter 4

The design of Clinical Documents Classification model

Contents

4.1 Introduction 119

4.2 The Description of Ohsumed Dataset 120

4.3 Proposed Models 121

4.3.1 Numerical Text Representation Methods 123

4.3.2 Artificial Neural Network (ANN) Architectures 124

4.4 Results and Discussion 124

4.5 Conclusion 127

4.1 Introduction

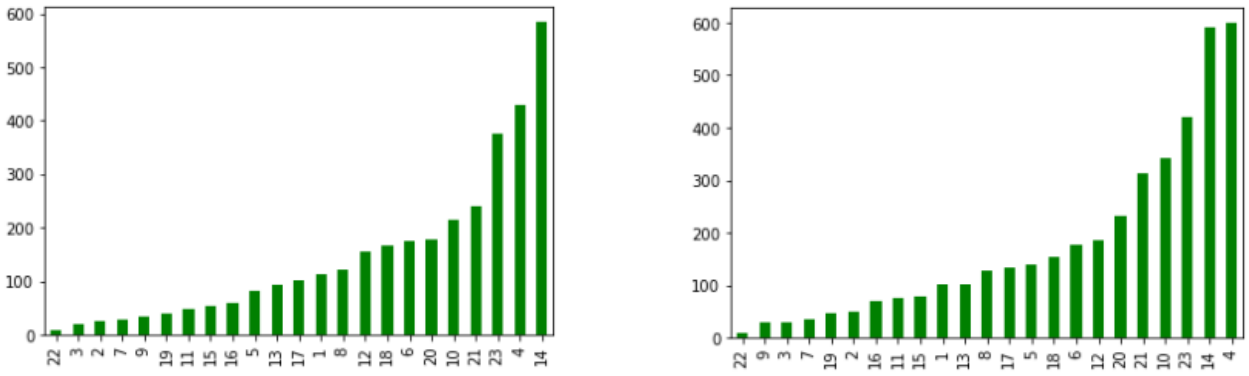
Accurate categorization of narrative medical texts into well defined disease classes is an essential prerequisite for large scale clinical research, automated decision support, and evidence based healthcare management. Despite the growing availability of electronic repositories, such as MEDLINE and institutional EMRs, most clinical texts remain unstructured, making their systematic exploitation a non trivial task. This chapter therefore investigates neural network techniques for medical texts classification, with a particular focus on transforming free text abstracts into reliable diagnostic labels.

4.2 The Description of Ohsumed Dataset

The Ohsumed dataset consists of abstracts categorized under Medical Subject Headings (MeSH) from the MEDLINE bibliographic database, covering medical literature from 1987 to 1991 and maintained by the National Library of Medicine (NLM). Most references originate from journal articles, conference proceedings, and reports [102].

The task involved classifying 23 disease categories. A total of 7,400 samples from the Ohsumed-20000 dataset were used, following the same dataset selection as in [199]. This dataset is publicly available at https://github.com/yao8839836/text_gcn. The same number of samples was also used in studies [105], [106], and [200]. To ensure a balanced dataset, 75% (5,550 samples) were allocated for training and 25% (1,850 samples) for testing, as presented in Figure 4.1.

Designation of the disease	Label
Bacterial Infections and Mycoses	1
Virus Diseases	2
Parasitic Diseases	3
Neoplasms	4
Musculoskeletal Diseases	5
Digestive System Diseases	6
Stomatognathic Diseases	7
Respiratory Tract Diseases	8
Otorhinolaryngologic Diseases	9
Nervous System Diseases	10
Eye Diseases	11
Urologic and Male Genital Diseases	12
Female Genital Diseases and Pregnancy Complications	13
Cardiovascular Diseases	14
Hemic and Lymphatic Diseases	15
Neonatal Diseases and Abnormalities	16
Skin and Connective Tissue Diseases	17
Nutritional and Metabolic Diseases	18
Endocrine Diseases	19
Immunologic Diseases	20
Disorders of Environmental Origin	21
Animal Diseases	22
Pathological Conditions, Signs and Symptoms	23



(a) Train data.

(b) Test data.

Figure 4.1: Distribution of training and testing datasets (Ohsumed).

4.3 Proposed Models

The Ohsumed_20000 dataset originally contained 23,166 documents. After identifying and removing duplicate reports, the dataset was reduced to 13,929 unique documents. A similar filtering process was applied to the dataset used for model training, ensuring that the 7,400 documents selected contained no duplicates.

Following this, the text was standardized by converting all characters to lowercase, and two vectorization approaches were implemented (Algorithm 7). In the first approach, the vocabulary was derived from the entire Ohsumed_20000 corpus, resulting in 44,191 tokens. In the second approach, the vocabulary was restricted to the training dataset of 7,400 documents, producing 34,523 tokens. The model trained using the vocabulary from the full corpus is denoted VNN, while the model trained with the vocabulary specific to the training dataset is denoted SNN.

Both models VNN and SNN were trained using four text vectorization methods: TF, BOW, TF-IDF, and DTM. Training was conducted for each case over 16, 32, and 64 epochs, with three runs performed for each configuration.

Algorithm 7 Pseudo algorithm for medical texts classification

Begin
Collect documents for training and testing datasets
Merge train and test datasets: `Data1 = Ohsumed-20000`, `Data2 = 7,400 samples`
Remove duplicate documents from Data1 and Data2
Split Data2: 75% for training, 25% for testing
Convert all text to lowercase
Transform class labels into categorical format
Define `Set_epochs` = {16, 32, 64}
Define `Set_models` = {SNN, VNN}
    SNN uses vocabulary of Data2 for vectorization
    VNN uses vocabulary of Data1 for vectorization
Define `Set_Vectorization` = {TF, DTM, TF-IDF, BOW}
for model in `Set_models` **do**
    for epoch in `Set_epochs` **do**
        for vect in `Set_Vectorization` **do**
            Train model
            Test model
        end for
    end for
end for
`Best_model` = max(train accuracy, test accuracy)
End

4.3.1 Numerical Text Representation Methods

Bag of Words (BOW) Representation: The BOW model represents text as a binary vector, where each word in a predefined vocabulary is mapped to a position in the vector. Given a dictionary W, where $W = \{w_1, w_2, ..., w_n\}$ and each w_i represents a word, the vocabulary is constructed from the entire corpus.

In this study, the vocabulary size differs between the two models:

- **SNN model:** 34,523 words
- **VNN model:** 44,191 words

Each document is represented as a vector $d = (d_1, d_2, ..., d_n)$, where:

- $d_i = 1$ if the word w_i appears in the document.
- $d_i = 0$ if the word w_i is absent.

Document Terme Matrix (DTM): A DTM is a structured representation of a text corpus in matrix form. In this matrix, each row corresponds to a document, and each column represents a unique token (word) from the vocabulary. The value at position M_{ik} indicates how many times the token in column k appears in the document represented by row i. This format allows for efficient text analysis and retrieval by transforming textual data into a numerical representation.

Term Frequency (TF): The TF of a word in a document measures how often the word appears relative to the total number of words in that document. It is calculated using the formula (4.1):

$$tf(t, d) = \frac{f_{t,d}}{\text{wc}_d} \tag{4.1}$$

where:

- $f_{t,d}$ represents the number of times the term t appears in document d.
- wc_d denotes the total number of words in document d.

Term Frequency Inverse Document Frequency (TF-IDF): The TF-IDF is a vectorization technique that combines Term Frequency (TF)—the number of times a term appears in a document—and Inverse Document Frequency (IDF), which measures how important a term is across the entire document collection.

The Inverse Document Frequency (IDF) is calculated as:

$$IDF(t) = \log \frac{N}{n} \tag{4.2}$$

where:

- N is the total number of documents in the collection.
- n is the number of documents in which the term appears.

Using both equations, the TF-IDF value for a term t_i is computed as:

$$TF\text{-}IDF = tf(t,d) \times IDF(t) = \frac{f_{t,d}}{\text{wc}_d} \times \log \frac{N}{n} \tag{4.3}$$

Equation (4.3) represents the TF-IDF formula, which is the product of TF and IDF. The term frequency adjusted for document length is represented by $\frac{f_{t,d}}{\text{wc}_d}$. Where $f_{t,d}$ indicates the number of times that a term t occurs in document d, and wc_d presents the number of words in document d. This technique helps to assign higher importance to words that appear frequently in a document but are rare across the entire dataset.

4.3.2 Artificial Neural Network (ANN) Architectures

The proposed ANN architecture is based on a Multilayer Perceptron (MLP), where data flows from an input layer to an output layer through one or more hidden layers. Unlike a simple perceptron, which has only one layer between input and output, the MLP structure allows for complex pattern recognition through these additional hidden layer (Table 4.1).

In these architectures:

- The input layer consists of k inputs, where:
  - $k = 34,523$ for the SNN model (corresponding to the vocabulary size for that model).
  - $k = 44,191$ for the VNN model (using a larger vocabulary size).
- The hidden layer contains 100 neurons.
- The output layer has 24 neurons, (To handle potential out-of-distribution or ambiguous cases, an additional output neuron was added, representing an 'unknown' class).

This structure enables the ANN to effectively learn and generalize from textual data.

4.4 Results and Discussion

Medical text classification plays a crucial role in modern healthcare by enabling the automatic categorization of narrative medical texts into specific disease categories. With the

Table 4.1: The architecture of the models SNN and VNN.

Layer (type)	Output Shape	Param #
dense_2 (Dense)	(None, 100)	3,452,400
dense_3 (Dense)	(None, 24)	2,424
Total params:		3,454,824
Trainable params:		3,454,824
Non-trainable params:		0

(a) SNN Model Summary

Layer (type)	Output Shape	Param #
dense_24 (Dense)	(None, 100)	4,419,200
dense_25 (Dense)	(None, 24)	2,424
Total params:		4,421,624
Trainable params:		4,421,624
Non-trainable params:		0

(b) VNN Model Summary

increasing volume of unstructured data generated in healthcare, such as medical reports, patient histories, and clinical notes, effectively organizing and extracting meaningful insights from this data has become essential. A reliable medical text classification system can bridge the gap between raw medical data and actionable information, providing healthcare professionals with tools to enhance diagnosis, treatment, and patient care.

One of the primary benefits of medical text classification is its ability to improve diagnostic efficiency. By automatically categorizing medical narratives into predefined disease categories, such as the 23 disease classes in this study, the system can assist clinicians in identifying relevant patterns and narrowing down possible diagnoses. This capability not only saves time but also reduces the likelihood of diagnostic errors, especially in complex cases where overlapping symptoms may obscure the correct diagnosis. Furthermore, automated classification can serve as a second opinion, complementing a clinician's judgment and reinforcing confidence in decision-making.

Medical texts are converted into numerical values using four encoding techniques: TF, TF-IDF, BOW, and DTM. The effectiveness of these vectorization methods are compared. Experimental results show that the proposed ANN architectures, namely the Simple Neural Network (SNN) and Vocabulary Neural Network (VNN), achieved notable performance for all four encoding methods. The SNN model utilizes a local vocabulary from 7,400 documents, while the VNN model leverages a global terminology dataset (Ohsumed_20000). Results indicate that the SNN model outperformed the VNN model in classification accuracy. Specifically, the SNN achieved an accuracy of 70.32% using TF with 64 epochs in the third run, 68.16% using BOW with 16 epochs in the first run, and 70.65% using DTM with 32 epochs in the third run. The highest accuracy, 71.08%, was obtained with TF-IDF using 16 epochs in the first run, demonstrating the superior performance of the SNN model for this encoding method, Table 4.2 summarizes the best results achieved.

Figure 4.2 presents the accuracy results of several studies in a specific domain, offering a comparative overview of the performance of different models. Among the studies listed, the results vary significantly, ranging from 36.2% to 72.8%, reflecting a considerable diversity in

Table 4.2: Synthesis of the best results diseases classification.

Encoding Method	Result
BOW	68.16% test accuracy with an epoch number 16 in time 1.
TF	70.32% test accuracy in time 3 with an epoch number of 64.
DTM	70.65% test accuracy with 32 epochs in time 3.
TF-IDF	71.08% test accuracy with 16 epochs in time 1.

model performance. At the top end of the spectrum, the study by [105] achieves the highest accuracy, reporting 72.8%, which sets a high benchmark for the other models.

In contrast, the lowest accuracy observed is from the study by [101], with a result of only 36.2%. This substantial gap between the highest and lowest accuracies indicates a significant variability in the outcomes of the different models. Several other studies report accuracy results within a narrow range of 68% to 69%, such as those by [106] (69.4%), [201] (68.5%), [200] (68.5%), and [199] (68.36%). These studies represent a solid foundation of performance but do not demonstrate the same level of distinction as the highest-performing study. In comparison, the accuracy of our model, reported as 71.08%, places it close to the top of the list. While it does not outperform the study by [105], it shows competitive performance, positioning it just below the highest accuracy observed. This suggests that our model is quite effective, with an accuracy that is close to the best-performing models in the field.

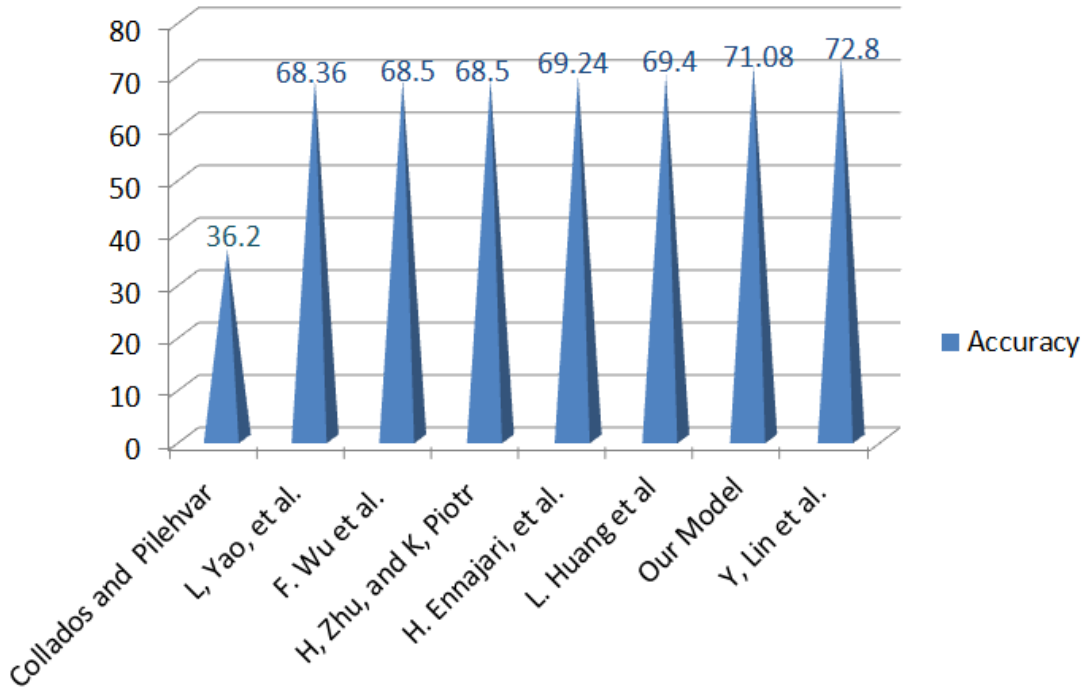


Figure 4.2: Performance comparison with previous studies for diseases classification.

Medical text classification is also invaluable for large-scale epidemiological studies and public health initiatives. By categorizing patient records, researchers can identify trends in disease prevalence, track outbreaks, and assess the impact of interventions across diverse populations. For example, categorizing medical texts into specific diseases enables researchers to analyze the incidence of chronic conditions like diabetes or cardiovascular diseases, as well as track emerging infectious diseases. These insights can inform healthcare policy, resource allocation, and preventive strategies, ultimately improving community health outcomes.

Moreover, automated classification systems can enhance the integration of EHRs. By processing unstructured narrative text and associating it with standardized disease categories, such systems ensure better data consistency and accessibility. This facilitates smoother communication between healthcare providers, insurers, and researchers, fostering a collaborative healthcare ecosystem. It also empowers patients, as more accurate and organized data can lead to personalized treatment plans and improved overall care.

In conclusion, medical text classification represents a vital advancement in leveraging healthcare data for improved clinical outcomes, research, and public health. By transforming narrative medical texts into structured disease categories, these systems can address challenges of scalability and complexity in the healthcare industry, supporting clinicians and researchers in delivering efficient, evidence-based, and patient-centered care.

4.5 Conclusion

This chapter investigated medical text classification using narrative medical text as input, aiming to predict the disease category among 23 predefined classes. The Ohsumed dataset served as the foundation for our analysis, providing a rich collection of medical texts. We employed several text vectorization techniques, including BOW, DTM, TF, and TF-IDF, to preprocess the text and train classification models. Our results demonstrate that the choice of vectorization method significantly impacts classification performance. The findings highlight SNN using TF-IDF with 16 epochs in time 1 achieved the best results, indicating its suitability for capturing the nuanced features of medical text. By systematically evaluating models trained with different configurations, we showcased the potential of combining traditional text preprocessing techniques with robust classification models to achieve accurate disease categorization. These insights emphasize the study's contribution to advancing the field of medical text classification.

The reliance on traditional vectorization techniques, such as BOW and TF-IDF, may not fully capture the semantic relationships within the text, limiting the model's ability to generalize effectively to unseen data.

This study [202] ontributes to the field of NLP in healthcare by systematically evaluating the effectiveness of various vectorization techniques for medical text classification. The results provide valuable insights into the strengths and limitations of traditional methods like BOW and TF-IDF when applied to the domain of medical narratives. This approach enriches the existing literature by offering empirical evidence on the applicability of these approaches to disease categorization tasks. Practically, the findings have significant implications for the development of automated systems in healthcare. Accurate classification of medical narratives into disease categories has the potential to streamline clinical workflows, improve decision support systems. For instance, an automated system based on the findings of this study could assist in efficiently categorizing unstructured medical notes, thereby enabling faster diagnosis and treatment planning. By bridging the gap between unstructured data and actionable insights, this research underscores the transformative potential of NLP in healthcare. Future work should explore integrating more advanced techniques, such as word embeddings (e.g., Word2Vec, GloVe) or contextual language models (e.g., BERT, GPT), which are designed to capture deeper semantic and contextual nuances in text.

Chapter 5

A novel framework for Detection of Drugs Efficiency

Contents

5.1 Introduction 129

5.2 Used Dataset 130

5.3 Data Preprocessing Strategies 130

5.3.1 First Strategy: Minimal Text Transformation 130

5.3.2 Second Strategy: Lemmatization Approach 131

5.4 Numerical Text Representation 132

5.5 Proposed Models 132

5.5.1 Traditional Machine Learning Models 132

5.5.2 Neural Network Models 132

5.6 Results and Discussion 135

5.6.1 Analysis of Model Performance without Lemmatization 135

5.6.2 Analysis of Model Performance with Lemmatization 136

5.7 Conclusion 139

5.1 Introduction

This chapter explores the automatic detection of drug effectiveness based on patient reviews of Drugs.com dataset. It presents two text preprocessing strategies—minimal transformation

and lemmatization—and examines their impact alongside sampling techniques (random and balanced) and different data split ratios (75/25 and 80/20). Various vectorization methods are used to encode the data for ML. The chapter evaluates both traditional classifiers and ANN models incorporating embedding and convolutional layers. Results show that model performance is strongly influenced by preprocessing, sampling strategy, and model complexity. The study underscores the potential of NLP-based methods for supporting drug evaluation and personalized medicine.

5.2 Used Dataset

drugs.com dataset includes 215,063 samples presented in two files (training and test), extracted from the UCI Machine Learning Repository[1]. this dataset is organized as follows: drug name and a drug's number, terms that describe the reason for using the drug or the patient's illness, user reviews, ratings given by the user, the date on which the drug was reviewed, and the number of users who found the review useful. The two files—drug_train (161,297 samples) and drug_test (53,766 samples)—were merged and the training and testing datasets have been reconstructed by randomly selecting documents for two case studies. In the first case, we allocated 75% of the data for training and 25% for testing. In the second case, we divided the data into 80% for training and 20% for testing. Additionally, we implemented both random sampling (Figure 5.1a) and random balanced sampling (Figure 5.1b) techniques as is detailled in [157].

5.3 Data Preprocessing Strategies

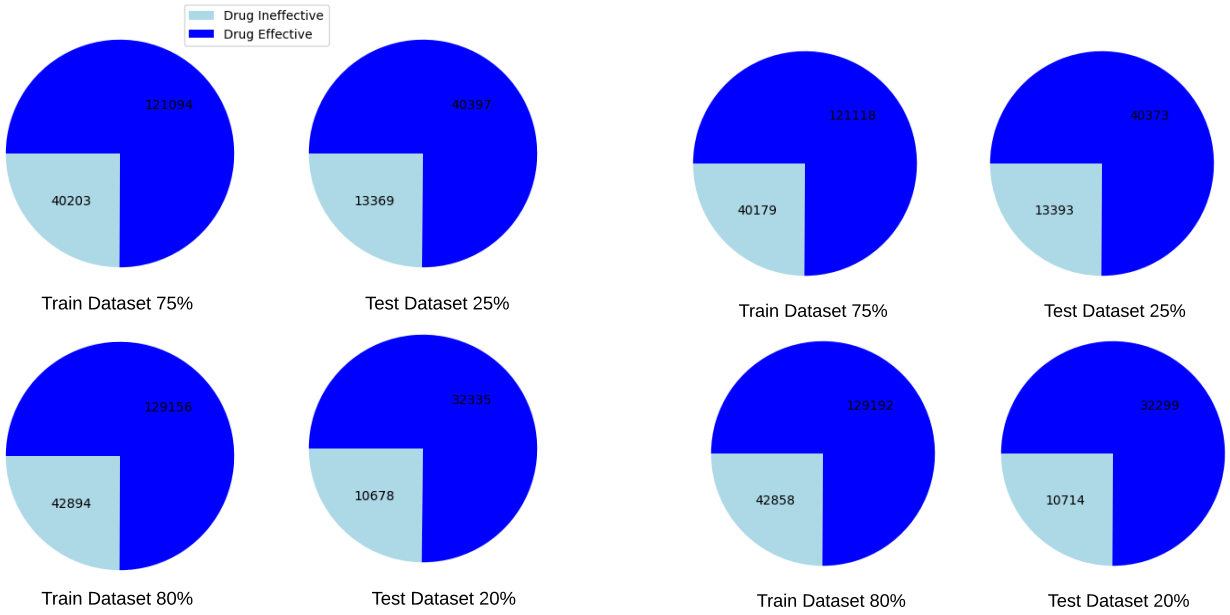
To prepare text data for analysis, we applied two different preprocessing strategies.

5.3.1 First Strategy: Minimal Text Transformation

In the first strategy, we aimed to preserve the original meaning of the text while making minimal modifications. The steps included:

- Converting all uppercase letters to lowercase.
- Eliminating extra spaces in drug names and medical conditions to ensure they are treated as a single token.
- No stemming or lemmatization was applied to preserve the full meaning of words.

[1] https://archive.ics.uci.edu/ml/datasets/Drug+Review+Dataset+%28Drugs.com%29



(a) Random sampling.

(b) Balanced random sampling.

Figure 5.1: Train and test datasets for Drugs.com corpus after preprocessing for random sampling and balanced random sampling strategies.

- No removal of punctuation marks or numbers, as elements like ”125 mg” or ”300 mg” provide crucial medical information.

This approach helps retain the context of why a patient is taking a specific medication and how they react to it. Additionally, avoiding stemming prevents expressions like *“taking medication,” “takes medication,” and “take medication”* from being erroneously conflated into a single root form, thereby preserving the distinct semantic nuances of each phrase.

After applying this strategy, the final vocabulary size is 84,362 unique words, and the document length is 2,405 words.

5.3.2 Second Strategy: Lemmatization Approach

In the second approach, we applied lemmatization using the NLTK library with the WordNet dictionary. This method normalizes words while preserving their meaning.

- Unlike stemming, lemmatization considers the context of words, preventing semantic distortions.

This strategy resulted in a slightly smaller vocabulary size (83,162 words), but the document length remained unchanged at 2,405 words.

5.4 Numerical Text Representation

Once the text was preprocessed, it needed to be converted into a numerical format for ML models.

- Each instance—consisting of the medication name, associated clinical condition, and patient commentary—was combined and converted into a numerical feature vector.
- Since different vectors have varying lengths, we applied padding to standardize them to a fixed size of 2,405.
- For statistical classifiers, we used one-hot encoding.
- For ANN models, we tested both one-hot encoding and word embeddings[2] with dimensions 32 and 100, to analyze how different vectorization techniques impact model performance.

5.5 Proposed Models

Several ML and neural network models were tested for classification tasks.

5.5.1 Traditional Machine Learning Models

Among probabilistic models, the following classifiers were employed using the scikit-learn library:

- **Logistic Regression**; he max_iter parameter was specified as 10, defining the upper limit for iterations performed by the solver during the training phase.
- **Random Forests**: 10 decision trees (n_estimators=10) with a maximum depth of 3 (max_depth=3).
- **Decision Trees**: Limited to a maximum depth of 3.
- **K-Nearest Neighbors (KNN)**: Using Euclidean distance with $k = 1$.

5.5.2 Neural Network Models

The study also explored DL models using Keras' Sequential API, which includes an embedding layer to transform words into high-dimensional numerical vectors for improved semantic representation.

[2] https://keras.io/guides/

Each document containing a drug name, condition, and patient comment was initially represented using one-hot encoding, later transformed into a dense numerical vector through an embedding layer. The following neural models were proposed:

- **Embedding-100**: A simple perceptron with a 100-dimensional embedding layer. Each word is mapped to a vector of length 100, forming document representations of shape $(2405, 100)$. A flattening operation then converts this into a single vector of dimension (240500) for fully connected layers.
- **Embedding-32**: Similar to Simple10 (a network with 10 hidden neurons), but with an additional 32-dimensional embedding layer.
- **Embedding-32-MaxPooling**: Applies Global Max-Pooling on Embedding-32, reducing dimensionality by selecting the most significant features.
- **CNN-MaxPooling**: A CNN with:
  - Conv1D layer (128 filters, kernel size = 5).
  - Embedding of size 32.
  - GlobalMaxPool1D, which extracts the most relevant features from each embedding.

Table 5.1: Summary of Simple10 model architecture.

Layer (type)	**Output Shape**	**Param #**
dense (Dense)	(None, 10)	24,060
dense_1 (Dense)	(None, 1)	11
Total params:		24,071
Trainable params:		24,071
Non-trainable params:		0

Table 5.2: Summary of the Embedding-32- model architecture.

Layer (type)	**Output Shape**	**Param #**
embedding_1 (Embedding)	(None, 2405, 32)	3,118,912
flatten_1 (Flatten)	(None, 76960)	0
dense_3 (Dense)	(None, 10)	769,610
dense_4 (Dense)	(None, 1)	11
Total params:		3,888,533
Trainable params:		3,888,533
Non-trainable params:		0

For all neural models:

- The ReLU activation function was used for hidden layers.
- The sigmoid function was applied to the output layer.
- The Adam optimizer was employed for training.
- The loss function used was binary cross-entropy, and model performance was evaluated using accuracy.

The proposed models are organized into two main categories based on their architecture and implementation libraries. Additionally, the Tables 5.1, 5.2, 5.3, 5.4, and 5.5 provide a detailed architecture of the different neural network architectures studied.

Table 5.3: Summary of the Embedding-32-Maxpooling model architecture.

Layer (type)	**Output Shape**	**Param #**
embedding (Embedding)	(None, 2405, 32)	3,118,912
global_max_pooling1d (GlobalMaxPooling1D)	(None, 32)	0
dense (Dense)	(None, 10)	330
dense_1 (Dense)	(None, 1)	11
Total params:		3,119,253
Trainable params:		3,119,253
Non-trainable params:		0

Table 5.4: Summary of the Embedding-100 model strcture.

Layer (type)	**Output Shape**	**Param #**
embedding (Embedding)	(None, 2405, 100)	9,746,600
flatten (Flatten)	(None, 240500)	0
dense_2 (Dense)	(None, 1)	240,501
Total params:		9,987,101
Trainable params:		9,987,101
Non-trainable params:		0

Table 5.5: Summary of the CNN-Maxpooling model strcture.

Layer (type)	**Output Shape**	**Param #**
embedding_1 (Embedding)	(None, 2405, 32)	3,118,912
conv1d (Conv1D)	(None, 2401, 128)	20,608
global_max_pooling1d_1 (GlobalMaxPooling1D)	(None, 128)	0
dense_2 (Dense)	(None, 10)	1,290
dense_3 (Dense)	(None, 1)	11
Total params:		3,140,821
Trainable params:		3,140,821
Non-trainable params:		0

5.6 Results and Discussion

In this section, we compare the classification performance of our models under two distinct preprocessing regimes: with and without lemmatization. For each case, we evaluate four sampling configurations—random sampling and balanced random sampling—each applied with two train/test splits (75%/25% and 80%/20%). Our goal is to determine how medical text processing via lemmatization interacts with different sampling strategies and data partitions to influence model accuracy.

5.6.1 Analysis of Model Performance without Lemmatization

Table 5.6 presents the accuracy of different models trained without lemmatization, evaluated using two sampling strategies: random sampling and balanced random sampling. Each strategy was tested with two different train-test splits (75%-25% and 80%-20%).

The traditional ML models, such as LR, RF, and DT, show consistent accuracy across all configurations, with minor variations. Where, their test accuracy remains around 75%. KNN, on the other hand, exhibits high training accuracy (close to 1.0) but significantly lower test accuracy (around 79-80%).

The ANN-based models demonstrate superior performance compared to probabilistic models. The Embedding-100 model consistently achieves the highest accuracy (99.83%) across all train-test splits and sampling strategies, making it the most reliable model in this setting.

The Embedding-32 and Embedding-32-Maxpooling models also perform well, with test accuracy ranging from 85.9% to 90.8%. These results indicate that adding max-pooling helps improve generalization.

CNN-Maxpooling achieves an even higher test accuracy, reaching approximately 93.6%.

- Traditional models (Logistic Regression, Decision Tree, and Random Forest) maintain stable but relatively low accuracy (75%).
- KNN overfits the training data, as seen in its near-perfect training accuracy but significantly lower test accuracy.
- ANN models, particularly Embedding-100 and CNN-Maxpooling, outperform probabilistic models, demonstrating the effectiveness of DL for this task.
- The max-pooling does not contribute to enhanced embedding robustness, as observed in the Embedding-32-Maxpooling model relative to Embedding-32.

Overall, the results suggest that DL models, especially Embedding-100 and CNN-Maxpooling, are the most effective choices for predicting drug effectiveness from user re-

Table 5.6: Performance of models without applying lemmatization.

Models	Random sampling				Balanced random sampling			
	75% train 25% test		80% train 20% test		75% train 25% test		80% train 20% test	
	Train	Test	Train	Test	Train	Test	Train	Test
LogisticRegression	0.7508	0.7512	0.7506	0.7521	0.7509	0.7509	0.7509	0.7509
RandonForest	0.7508	0.7512	0.7506	0.7521	0.7509	0.7509	0.7509	0.7509
DecisionTree	0.7525	0.7534	0.7506	0.7521	0.7509	0.7509	0.7509	0.7509
KNeighbors	0.9960	0.7928	0.9986	0.8037	0.9986	0.7938	0.9986	0.8032
Embedding-100	**0.9983**	**0.9983**	**0.9983**	**0.9983**	**0.9984**	**0.9984**	**0.9984**	**0.9984**
Simple10	0.7508	0.7512	0.7506	0.7521	0.7509	0.7509	0.7509	0.7509
Embedding-32-	0.9972	0.9011	0.9976	0.9083	0.9961	0.8937	0.7509	0.7509
Embedding-32-Maxpooling	0.9176	0.8591	0.9236	0.8703	0.9152	0.8638	0.9179	0.8664
CNN-Maxpooling	0.9971	0.9322	0.9973	0.9365	0.9972	0.9340	0.9970	0.9347

views.

5.6.2 Analysis of Model Performance with Lemmatization

Table 5.7 presents the accuracy of different models trained with lemmatization. The models were evaluated under two sampling strategies: random sampling and balanced random sampling, each tested with two different train-test splits (75%-25% and 80%-20%).

The performance of traditional models, such as LR, RF, and DT, remains relatively stable, with test accuracy hovering around 75%. Compared to the results without lemmatization (Table 5.6), there is no significant improvement, indicating that lemmatization has little effect on these models.

KNN, still shows a significant gap between training accuracy (close to 1.0) and test accuracy (77.8%–80.5%). However, there is a slight decrease in test accuracy compared to the non-lemmatized case, suggesting that the removal of word variations may have led to information loss.

ANN models continue to outperform traditional models, with lemmatization showing some influence on performance:

- Embedding-100 achieves a strong performance with test accuracy ranging from 90.1% to 91.0%, slightly lower than the non-lemmatized version.
- Embedding-32 maintains high accuracy, but its test performance varies slightly, indicating sensitivity to lemmatization.
- Embedding-32-Maxpooling improves its test accuracy, reaching 86.8%, suggesting that max-pooling helps retain important contextual information even after lemmatization.
- CNN-Maxpooling remains one of the best-performing models, with test accuracy ranging from 92.7% to 93.8%. The results indicate that convolutional structures effectively mitigate potential information loss caused by lemmatization.

- Traditional models (LR, DT, and RF) are largely unaffected by lemmatization.
- KNN experiences a minor drop in accuracy, highlighting the sensitivity of distance-based methods to lemmatization.
- Deep learning models, particularly Embedding-100 and CNN-Maxpooling, still outperform traditional classifiers, though lemmatization slightly affects performance.
- Max-pooling strategies do not enhance the robustness of embeddings, as evidenced by the performance of the Embedding-32-Maxpooling and Embedding-32 configurations.

Overall, lemmatization has a limited impact on traditional models but slightly affects DL models. Despite this, CNN-based models continue to demonstrate the highest predictive power.

Table 5.7: Performance of models without applying lemmatization.

Models	Random sampling				Balanced random sampling			
	75% train 25% test		80% train 20% test		75% train 25% test		80% train 20% test	
	Train Acc	Test Acc	Train	Test	Train	Test Acc	Train	Test
LogisticRegression	0.7508	0.7512	0.7481	0.7491	0.7509	0.7509	0.7508	0.7508
RandonForest	0.7508	0.7512	0.7506	0.7521	0.7509	0.7509	0.7509	0.7509
DecisionTree	0.7537	0.7541	0.7506	0.7521	0.7509	0.7509	0.7523	0.7521
KNeighbors	0.9960	0.7782	0.9986	0.8054	0.9986	0.7745	0.9986	0.7924
Embedding-100	0.9982	0.9015	0.9984	0.9104	0.9984	0.9031	0.9983	0.9082
Simple10	0.7508	0.7512	0.7506	0.7521	0.7509	0.7509	0.7509	0.7509
Embedding-32-	0.9974	0.8996	0.7506	0.7521	0.9972	0.8966	0.9968	0.9026
Embedding-32-Maxpooling	0.9093	0.8610	0.9091	0.8626	0.9122	0.8632	0.9181	0.8683
CNN-Maxpooling	0.9960	0.9271	0.9962	0.9345	0.9963	0.9307	0.9965	0.9385

Figure 5.2 presents comparative analysis of various studies highlights the progression of studies and their corresponding accuracies in text classification tasks. These studies, spanning from 2015 to 2022, illustrate the growing effectiveness of modern techniques over time. In 2015, Na, Jin-Cheon, and Wai Yan Min Kyaing [203] achieved an accuracy of 0.69 using a linguistic-based method. While fundamental, this approach lacks the sophistication and adaptability of more advanced techniques, resulting in moderate performance. In contrast, Gräber et al. (2018) [97] employed logistic regression, a traditional ML method, and achieved a significantly higher accuracy of 0.9224, demonstrating the potential of simple yet robust models in specific applications. The use of DL methodologies began to gain traction in 2020. Bemila et al. [99] utilized a recurrent neural network with bidirectional long short-term memory (RNN-BiLSTM) and achieved an accuracy of 0.83906, effectively capturing sequential dependencies in text. Similarly, C. Colón-Ruiz and I. Segura-Bedmar [96] implemented a long short-term memory (LSTM) model, which outperformed its predecessors with an accuracy of 0.9046, showcasing its capability to manage long-term dependencies in data.

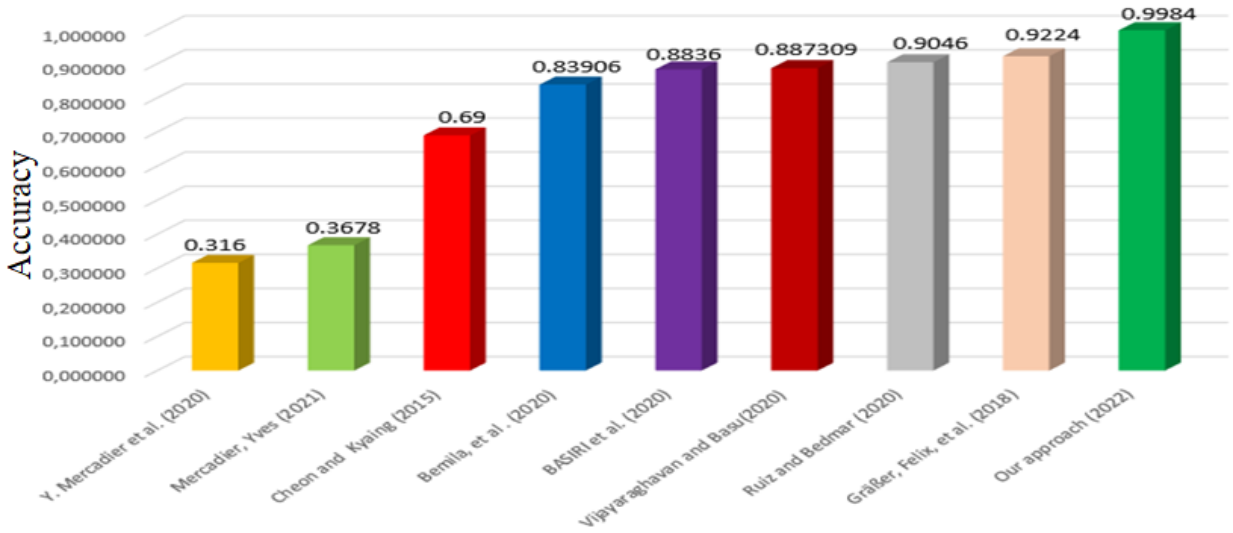


Figure 5.2: Comparison with previous studies. The result 0.9984 for train and test accuracy is obtained by Embedding-100 using 80% train 20% test and 75% train 25% test balanced random sampling without lemmatization.

Hybrid approaches also emerged in 2020, as exemplified by M.E. Basiri et al. [204], who combined wavelet transform and Naive Bayes (3W3DT-NB) to achieve an accuracy of 0.8836. Meanwhile, S. Vijayaraghavan and D. Basu [205] demonstrated the power of ANN, achieving a comparable accuracy of 0.887309. However, some studies reported lower performance with transformer-based models. For instance, Y. Mercadier et al. [98] applied Roberta with the Data Augmentation method (DAIA) but achieved 0.316 accuracy. Similarly, Mercadier, Yves (2021) [206] employed XLNet and reported an accuracy of 0.3678. These results suggest the need for better domain-specific adaptation or optimization when using advanced transformer architectures. Our study, conducted in 2022, introduces the Embedding-100 method, achieving a state-of-the-art accuracy of 0.9984. This approach leverages word embeddings to preserve semantic relationships and effectively captures contextual nuances, significantly outperforming all previous methods. In conclusion, the comparison reveals that while traditional and hybrid methods have their strengths, modern DL techniques, particularly those utilizing embeddings, are setting new benchmarks in accuracy. The steady progression of results underscores the importance of innovative methodologies in advancing text classification capabilities.

5.7 Conclusion

The Embedding-100 model was proposed to predict drug effectiveness and efficiency using NLP techniques. The study focused on addressing the critical challenge of capturing medical semantics to analyze and extract knowledge from medical text. The findings demonstrate that model accuracy is significantly influenced by the architecture, preprocessing method, and sampling type used. Experiments revealed that Embedding-100 achieved exceptional performance, with both training and test accuracies at 0.9984, outperforming other state-of-the-art models. The model integrates ANN, one-hot encoder, and word embedding to deliver superior results. While the Embedding-100 model has shown promising results, the current work is limited to a specific dataset, highlighting the need to extend the approach to other datasets to validate its robustness and adaptability. Future research should also explore enhancements to preprocessing techniques and examine the model's performance across diverse medical data types.

In conclusion, the Embedding-100 model contributes to advancing the theoretical understanding of NLP in the medical domain, particularly in handling medical semantics. By integrating ANN with one-hot encoding and word embedding, it provides a novel approach to improving model accuracy. Practically, this model offers a powerful tool for analyzing drug effectiveness and efficiency, with potential applications in healthcare decision-making, drug evaluation, and personalized medicine. Additionally, it can serve as a foundation for further research in medical text analysis and predictive modeling.

General Conclusion

Contents

Conclusion **140**

Future Works **141**

Valorization and Applications **143**

This chapter synthesizes the key findings of the thesis across three core domains: event, temporal expression, and temporal relation extraction; medical text classification; and drug efficiency prediction. It revisits the main contributions, highlights experimental results, and reflects on their implications for clinical NLP. The chapter also outlines future research directions, including ontology refinement, multilingual adaptation, and improved model interpretability. It concludes with a discussion on the real-world applications and impact of the proposed approaches in healthcare.

Conclusion

This thesis has addressed the challenges associated with extracting and analyzing knowledge from unstructured clinical narratives, with a specific focus on temporal events, entity recognition,temporal relation, medical text classification, and drug efficacy prediction. The research proposes an integrated framework that leverages both DL and ontological knowledge representation to enhance semantic understanding and improve the interpretability of clinical data.

The first major contribution is the development of the CONSEPTR model, a modular convolutional neural architecture incorporating standard, transposed, and separable convolutions, complemented by BERT contextual embeddings. This model has demonstrated strong performance in recognizing clinical events and temporal expressions, achieving consistent results across benchmark datasets (i2b2-2010, 2012, 2018). The use of diverse convolutional layers allowed the system to learn hierarchical and positionally aware features, making it well-suited for the complexity of narrative medical text.

To further enrich semantic reasoning, the CONSEPTR model was integrated with ontologies using both the Jena API and the Owlready2 library. This enabled the system to benefit from structured medical knowledge, improving entity disambiguation and extending the capacity to identify concepts not explicitly labeled in the original datasets. Notably, the Jena API facilitated SPARQL-based querying for real-time semantic inference, while Owlready2 provided a flexible and Python-compatible environment for ontology manipulation. This thesis has introduced an integrated, comprehensive approach for extracting temporal and relational knowledge from unstructured clinical narratives, addressing key challenges in clinical NLP. In addition the proposed model (R_CONSEPTRL_8) combines DL architectures—including various CNNs models (Standard Convolution, Transposed Convolution, Separable Convolution, and ConvLSTM). Particularly, the adoption of NNLM sentence embeddings demonstrated high effectiveness in capturing temporal information, especially for complex and lengthy clinical sentences typical of EHRs.

Furthermore the thesis explored medical text classification using the Ohsumed dataset, comparing different text vectorization methods (TF, TF-IDF, BOW, DTM) under two vocabulary scopes. The results demonstrated the effectiveness of ANNs (SNN) combined with TF-IDF, emphasizing the impact of vocabulary granularity and vectorization on classification performance.

A final application examined the prediction of drug efficacy using patient-generated reviews. Here, the Embedding-100 model, proved effective despite the linguistic variability in user-generated text. Experimental results highlighted the importance of preprocessing, sampling strategies, and model architecture in handling informal and unstructured medical commentary.

Taken together, these contributions underscore the value of combining data-driven learning with structured domain knowledge. The proposed approaches have shown promising generalization capabilities and demonstrated scalability across multiple clinical text tasks. The synergy between DL and ontologies offers a pathway toward more interpretable, accurate, and context-aware information extraction systems in healthcare.

Future Works

Several avenues for future research are identified, which can enhance the model's performance, applicability, and integration within the medical domain.

- **Temporal reasoning and causal inference:** Integrating more expressive narrative knowledge representation frameworks such as the Narrative Knowledge Representation Language (NKRL) may enable deeper semantic modeling of complex event structures

and temporal reasoning in clinical narratives. In addition beyond extracting temporal relations, future work could focus on causal relationship identification, enabling deeper clinical insight generation, such as predicting disease progression or treatment outcomes.

- **Expanding the analysis of drug effectiveness:** In the future, our work can be extended to analyze drug effectiveness across a broader range of datasets, particularly medical records. The ability to evaluate drug effectiveness using diverse data sources will enhance the generalizability and robustness of the model. We plan to investigate the model's performance on datasets containing real-world clinical data, which may present new challenges and offer more nuanced insights into drug efficacy in varied patient populations.
- **Enhancing medical text preprocessing:** Medical text preprocessing remains a critical research area that directly impacts the accuracy and efficiency of models analyzing clinical data. Future work will focus on refining preprocessing techniques, specifically addressing issues such as the normalization of medical terms, handling abbreviations, and improving the extraction of structured data from unstructured text. Developing more sophisticated preprocessing pipelines will be essential for handling the complexities and heterogeneity found in clinical narratives.
- **Medical text classification and semantic encoding:** For clinical text classification, we aim to explore methods to bring semantic understanding into the encoding models. Future studies will focus on how to improve the representation of semantic relationships within medical texts, enabling the model to better understand context and nuances. This will help achieve more accurate and interpretable classifications of medical texts, facilitating better clinical decision-making.
- **Ontology construction and customization:** Future research could focus on improving the design of ontologies by developing customized ones that are specifically tailored to the distinct features and nuances of medical text data. This personalized approach will allow for a more context-sensitive use of ontological knowledge, resulting in better semantic representation and more accurate entity extraction from biomedical narratives. Furthermore, we aim to explore relationships between entities to reveal hidden insights, enriching the overall knowledge extraction process. By incorporating more extensive knowledge bases and applying the methodology to a broader range of biomedical datasets, we can enhance its versatility and reliability.
- **Leveraging entity inferences for knowledge expansion:** Integrating inferences between entities is another promising direction for future research. By uncovering hidden relationships and patterns between entities, we can enrich the knowledge extracted

from medical texts, providing a deeper understanding of the data. Future work will focus on using reasoning and inference mechanisms to generate new insights that go beyond the initial entity extraction process.

- **Expanding to diverse biomedical datasets:** Finally, expanding the methodology to diverse biomedical datasets is a crucial area for future development. By applying the model to different types of biomedical data, we can test its robustness and adaptability. This expansion will include datasets from various domains, such as EHRs, clinical trials, and medical literature, further enhancing the applicability and generalization of the approach.

Valorization and Applications

Practically, the findings have significant implications for the development of automated healthcare systems. Accurate classification of unstructured medical narratives can streamline clinical workflows, enhance decision support systems, and improve the accessibility and organization of patient information. Systems informed by these findings could assist clinicians in categorizing large volumes of free-text medical notes more efficiently, ultimately supporting faster diagnosis, more effective treatment planning, and better patient outcomes. By bridging the gap between unstructured clinical data and actionable medical insights, this work underscores the transformative role of NLP in modern healthcare.

The works presented in this thesis has several significant implications for the field of medical text analysis, with broad potential for real-world applications and future advancements. By developing a robust models capable of extracting events and temporal expressions, classifying diseases, and predicting drug effectiveness from unstructured medical texts, this research contributes to the ongoing efforts in advancing healthcare informatics. The valorization of this work lies not only in the theoretical contributions but also in its practical applications, which can have transformative effects on clinical workflows, drug development, and decision support systems.

- **Enhancing clinical decision support systems:** The ability to classify medical texts accurately and extract critical information, such as temporal expressions and disease-related events, as well relational knowledge extracted from free-text clinical notes hold great promise for enhancing clinical decision support systems. These systems rely on the timely and accurate analysis of medical data to aid clinicians in diagnosing diseases and recommending treatments. The methods proposed in this study can be integrated into existing clinical platforms, facilitating the automatic classification of

medical texts, such as EHRs, thereby improving clinical decision-making and reducing the cognitive load on healthcare professionals.

- **Medical record summarization:** The temporal relation extraction approach can be applied to automatically generate temporal summaries of patient histories, aiding clinicians in rapidly understanding disease progression and treatment responses.
- **Improving drug evaluation and development:** The model's application to the analysis of drug effectiveness, particularly through patient reviews, offers substantial value in the pharmaceutical industry. By analyzing patient feedback, the model can contribute to evaluating the real-world effectiveness of drugs, identifying side effects, and improving patient care. Additionally, the integration of advanced NLP methods, such as word embeddings, enables more accurate insights from diverse and informal language in patient reviews. This application is not limited to post-market drug evaluation; it can also support drug development by uncovering hidden relationships between drugs and their effects.
- **Customized ontologies for domain-specific applications:** One of the key contributions of this study is the use of customized ontologies, particularly in the context of medical text data. Medical ontologies to reflect the specific characteristics and subtleties of medical texts allows for more accurate entity extraction and better context-aware utilization of knowledge. This approach can be applied across various domains within healthcare, such as medical research, clinical trial analysis, and patient care, ensuring that the information extracted from texts is semantically enriched and aligned with domain-specific needs. Further valorization can come from expanding this approach to other specialized fields, enhancing the overall efficiency and accuracy of information extraction systems.
- **Expanding to diverse biomedical datasets:** The model's potential applications extend beyond the datasets used in this thesis. By applying the methodology to a wider array of biomedical datasets, including electronic health records, clinical trials, and medical literature, the approach can be adapted to various healthcare settings, thereby enhancing its scalability and robustness. This expansion could significantly impact areas such as personalized medicine, where the integration of vast amounts of clinical data can lead to more tailored and effective treatment plans for patients.
- **Advancing research and development in medical NLP:** This work also contributes to the broader field of NLP in healthcare by offering new insights into the use of DL models and ontologies for medical text analysis. The proposed methods and results can inspire further research into integrating other DL models and enhancing existing ones. Additionally, by addressing the complexities of medical text preprocess-

ing, this study lays the groundwork for future advancements in the field, encouraging the development of more sophisticated and accurate NLP models for healthcare applications.

Bibliography

[1] A. Géron, *Hands-On Machine Learning With Scikit-Learn And Tensorflow: Concepts, Tools, And Techniques to Build Intelligent Systems.* O'Reilly Media, 2017.

[2] B. He, Y. Guan, and R. Dai, "Classifying medical relations in clinical text via convolutional neural networks," *Artificial Intelligence In Medicine*, vol. 93, p. 43–49, 2019.

[3] M. Nii, Y. Tsuchida, Y. Kato *et al.*, "Analysis of classification results for the nursing-care text evaluation using convolutional neural networks," in *2017 6th International Conference on Informatics, Electronics and Vision & 2017 7th International Symposium in Computational Medical and Health Technology (ICIEV-ISCMT).* IEEE, 2017, pp. 1–6.

[4] Y. Wang, S. Sohn, S. Liu, F. Shen, L. Wang, E. J. Atkinson, S. Amin, and H. Liu, "A clinical text classification paradigm using weak supervision and deep representation," *BMC Medical Informatics and Decision Making*, vol. 19, pp. 1–13, 2019. [Online]. Available: https://doi.org/10.1186/s12911-018-0723-6

[5] P. B. Jensen, L. J. Jensen, and S. Brunak, "Mining electronic health records: Towards better research applications and clinical care," *Nature Reviews Genetics*, vol. 13, no. 6, pp. 395–405, 2012.

[6] H. Dalianis, *Clinical Text Mining: Secondary Use of Electronic Patient Records.* Springer, 2018.

[7] A. Panesar, *Machine Learning and AI for Healthcare.* Springer, 2021.

[8] A. Saxena and S. Chandra, *Artificial Intelligence and Machine Learning in Healthcare.* Springer, 2021.

[9] R. Tibshirani, D. Witten, T. Hastie, and G. James, *An Introduction to Statistical Learning with Applications in R*, 2nd ed. Springer, 2017.

[10] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach.* Prentice Hall, 2010.

[11] T. M. Mitchell, *Machine Learning.* McGraw-Hill, 1997.

[12] Y. B. Ian Goodfellow and A. Courville, *Deep Learning.* MIT Press, 2016.

[13] L. Breiman, J. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, 1st ed. Chapman and Hall/CRC, 1984. [Online]. Available: https://doi.org/10.1201/9781315139470

[14] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, pp. 81–106, 1986.

[15] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[16] ——, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996.

[17] ——, "Pasting small votes for classification in large databases and online," *Machine Learning*, vol. 36, pp. 85–103, 1999.

[18] G. Louppe, "Understanding random forests: From theory to practice," Ph.D. dissertation, University of Liège, 2014. [Online]. Available: https://arxiv.org/abs/1407.7502

[19] B. Efron and R. Tibshirani, "An introduction to the bootstrap."

[20] T. G. Dietterich, "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization," *Machine Learning*, vol. 40, pp. 139–157, 2000.

[21] E. Fix and J. L. Hodges, "Discriminatory analysis: Nonparametric discrimination: Consistency properties," *Technical Report No. 4, USAF School of Aviation Medicine*, 1951.

[22] A. Saxena and S. Chandra, *Artificial intelligence and machine learning in healthcare*. Springer, 2021.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 25, 2012.

[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[25] A. Vaswani, N. Shazeer, and N. e. a. Parmar, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[26] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[28] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[29] T. Brown, B. Mann, and N. e. a. Ryder, "Language models are few-shot learners," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020.

[30] A. Esteva, B. Kuprel, and R. A. e. a. Novoa, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[31] M. Bojarski, D. Testa, and D. e. a. Dworakowski, "End-to-end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.

[32] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, pp. 386–408, 1958.

[33] R. J. W. David E. Rumelhart, Geoffrey E. Hinton, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533–536, 1986.

[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[35] Y. L. et al., "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, 1989.

[36] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[37] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[38] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung *et al.*, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[39] S. G. M. R. S. V. K. S. Jeff Donahue, Lisa Anne Hendricks and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[40] B. Smith, "Blackwell guide to the philosophy of computing and information," 2003.

[41] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[42] N. Guarino, *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy.* IOS press, 1998, vol. 46.

[43] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, no. 5, pp. 34–43, 2001.

[44] A. Borgida, "Description logics in data management," *IEEE Transactions on Knowledge and Data Engineering*, vol. 7, no. 5, pp. 671–682, 1995.

[45] R. Fikes and A. Farquhar, "Distributed repositories of highly expressive reusable ontologies," *IEEE Intelligent Systems*, vol. 14, no. 2, pp. 73–79, 1999.

[46] G. Correndo, M. Salvadores, I. Millard, H. Glaser, and N. Shadbolt, "Sparql query rewriting for implementing data integration over linked data," in *Proceedings of the 2010 EDBT/ICDT Workshops*, 2010, pp. 1–11. [Online]. Available: https://doi.org/10.1145/1754239.1754244

[47] M. Ashburner, C. A. Ball, J. A. Blake *et al.*, "Gene ontology: tool for the unification of biology," *Nature genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[48] A. Hogan, E. Blomqvist *et al.*, *Knowledge graphs*, 2021, vol. 54, no. 4.

[49] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[50] A. Pease, *Ontology: A practical guide.* Articulate Software Press, 2011.

[51] I. Horrocks, P. F. Patel-Schneider, and F. Van Harmelen, "From shiq and rdf to owl: The making of a web ontology language," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 1, no. 1, pp. 7–26, 2003.

[52] L. Stojanovic, A. Maedche, B. Motik, and N. Stojanovic, "User-driven ontology evolution management," in *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web: 13th International Conference, EKAW 2002 Sigüenza, Spain, October 1–4, 2002 Proceedings 13.* Springer, 2002, pp. 285–300.

[53] D. Kless and L. Janson, "How fit is owl to represent realist ontologies," *The semantics of representational units in realist ontologies and the Web Ontology Language*, 2014.

[54] P. Hitzler, M. Krötzsch, and S. Rudolph, *Foundations of Semantic Web Technologies*. Chapman and Hall/CRC, 2009.

[55] G. Antoniou and F. van Harmelen, *A Semantic Web Primer*. MIT Press, 2004.

[56] D. Allemang and J. Hendler, *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. Elsevier, 2011.

[57] A. Hogan, E. Blomqvist, M. Cochez, C. D'Amato, G. de Melo, C. Gutierrez, ..., and A. Zimmermann, "Knowledge graphs," *ACM Computing Surveys (CSUR)*, vol. 54, no. 4, pp. 1–37, 2021.

[58] I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen, "From shiq and rdf to owl: The making of a web ontology language," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 1, no. 1, pp. 7–26, 2003.

[59] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. F. Patel-Schneider, and U. Sattler, "Owl 2: The next step for owl," *Journal of Web Semantics*, vol. 6, no. 4, pp. 309–322, 2008.

[60] O. Erling and I. Mikhailov, "Rdf support in the virtuoso dbms," in *Networked Knowledge-Networked Media*. Springer, 2009, pp. 7–24.

[61] T. Heath and C. Bizer, *Linked Data: Evolving the Web into a Global Data Space*. Morgan and Claypool, 2011.

[62] T. Sigwele, A. Naveed, Y. F. Hu, M. Ali, J. Hou, M. Susanto, and H. Fitriawan, "Building a semantic restful api for achieving interoperability between a pharmacist and a doctor using jena and fuseki," in *Journal of Physics: Conference Series*, vol. 1376, no. 1. IOP Publishing, 2019, p. 012015.

[63] J.-B. Lamy, "Owlready: Ontology-oriented programming in python with automatic classification and high level constructs for biomedical ontologies," *Artificial intelligence in medicine*, vol. 80, pp. 11–28, 2017.

[64] ——, *Ontologies with python*. Apress, 2021.

[65] ——, *Ontologies avec Python: Programmer OWL 2.0 Ontologies avec Python et Owlready2*. APRESS, 2020.

[66] H. Tiaiba, L. Sabri, and O. Kazar, "Ontologies and machine learning models to enhance health informatics: A survey, challenges and future directions," *IAENG International Journal of Applied Mathematics*, vol. 55, no. 3, pp. 475–499, 2025.

[67] M. Jiang, Y. Chen, M. Liu, S. T. Rosenbloom, S. Mani, J. C. Denny, and H. Xu, "A study of machine-learning-based approaches to extract clinical entities and their assertions from discharge summaries," *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 601–606, 04 2011. [Online]. Available: https://doi.org/10.1136/amiajnl-2011-000163

[68] K. Huang, J. Altosaar, and R. Ranganath, "Clinicalbert: Modeling clinical notes and predicting hospital readmission," *arXiv preprint arXiv:1904.05342*, 2019.

[69] Y. Peng, Z. Zhang, X. Wang, L. Yang, and L. Lu, "Text mining and deep learning for disease classification," *Handbook of Medical Image Computing and Computer Assisted Intervention*, pp. 109–135, 2020.

[70] M. Bampa and H. Dalianis, "Detecting adverse drug events from swedish electronic health records using text mining," in *Proceedings of the LREC 2020 Workshop on Multilingual Biomedical Text Processing (MultilingualBIO 2020)*, 2020, pp. 1–8.

[71] M. Dinarelli and L. Grobol, "Modélisation d'un contexte global d'étiquettes pour l'étiquetage de séquences dans les réseaux neuronaux récurrents," in *Journée commune AFIA-ATALA sur le Traitement Automatique des Langues et l'Intelligence Artificielle pendant la onzième édition de la plate-forme Intelligence Artificielle (PFIA 2018)*, 2018.

[72] M. Hughes, I. Li, S. Kotoulas, and T. Suzumura, "Medical text classification using convolutional neural networks," in *Informatics for Health: Connected Citizen-Led Wellness and Population Health*, ser. Studies in Health Technology and Informatics, vol. 235. IOS Press, 2017, pp. 246–250.

[73] L. Yao, C. Mao, and Y. Luo, "Clinical text classification with rule-based features and knowledge-guided convolutional neural networks," *BMC Medical Informatics and Decision Making*, vol. 19, no. Suppl 3, p. 71, 2019. [Online]. Available: https://doi.org/10.1186/s12911-019-0781-4

[74] C. Cheng and J. C. Weiss, "Typed Markers and Context for Clinical Temporal Relation Extraction," *Proc Mach Learn Res*, vol. 219, pp. 94–109, 2023.

[75] C. Grouin, N. Grabar, T. Hamon, S. Rosset, X. Tannier, and P. Zweigenbaum, "Eventual situations for timeline extraction from clinical reports," *Journal of the American Medical Informatics Association*, vol. 20, no. 5, pp. 820–827, 2013.

[76] G. Moharasan and T. B. Ho, "Extraction of temporal information from clinical narratives," *Journal of Healthcare Informatics Research*, vol. 3, pp. 220–244, 2019.

[77] P. Li and H. Huang, "Deep learning based natural language processing system for clinical information identification from clinical notes and pathology reports," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*.

[78] Y. Long, Z. Li, X. Wang, and C. Li, "Xjnlp at semeval-2017 task 12: Clinical temporal information extraction with a hybrid model," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, 2017, pp. 1014–1018. [Online]. Available: https://doi.org/10.18653/v1/S17-2178

[79] Y.-K. Lin, H. Chen, and R. A. Brown, "Medtime: A temporal information extraction system for clinical narratives," *Journal of Biomedical Informatics*, vol. 46, pp. S20–S28, 2013.

[80] S. Zhang, Y. Li, and S. Li, "Bi-Lstm-Crf Network For Clinical Event Extraction With Medical Knowledge Features," *IEEE Access*, vol. 10, pp. 110 100–110 109, 2022.

[81] Y. Si, J. Wang, H. Xu, and K. Roberts, "Enhancing clinical concept extraction with contextual embeddings," *J Am Med Informatics Assoc*, vol. 26, no. 11, pp. 1297–1304, 2019.

[82] K. Roberts, B. Rink, and S. M. Harabagiu, "A flexible framework for recognizing events, temporal expressions, and temporal relations in clinical text," *Journal of the American Medical Informatics Association*, vol. 20, no. 5, pp. 867–875, 2013.

[83] C. Grouin, N. Grabar, T. Hamon *et al.*, "Eventual situations for timeline extraction from clinical reports," *Journal of the American Medical Informatics Association*, vol. 20, no. 5, pp. 820–827, 2013.

[84] S. Sohn, K. B. Wagholikar, D. Li *et al.*, "Comprehensive temporal information detection from clinical text: medical events, time, and tlink identification," *Journal of the American Medical Informatics Association*, vol. 20, no. 5, pp. 836–842, 2013.

[85] S. S. Islam and A. A. Hossain, "Glioma segmentation from multimodal mri using u-net convolutional neural networks," in *2021 IEEE 2nd International Conference on Applied Electromagnetics, Signal Processing, & Communication (AESPC)*. IEEE, 2021, pp. 1–5.

[86] Y. Zhou, Y. Yan, R. Han *et al.*, "Clinical temporal relation extraction with probabilistic soft logic regularization and global inference," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 16. AAAI, 2021, pp. 14 647–14 655.

[87] G. Alfattni, N. Peek, and G. Nenadic, "Attention-based bidirectional long short-term memory networks for extracting temporal relationships from clinical discharge summaries," *Journal Of Biomedical Informatics*, vol. 123, p. 103915, 2021.

[88] H. Haq, V. Kocaman, and D. Talby, "Deeper clinical document understanding using relation extraction," *arXiv preprint arXiv:2112.13259*, 2021.

[89] V. Kocaman and D. Talby, "Improving clinical document understanding on covid-19 research with spark nlp," in *Proceedings of the Workshop on Scientific Document Understanding co-located with 35th AAAI Conference on Artificial Intelligence*. AAAI, 2021, virtual event.

[90] N. Nath, S. Lee, M. McDonnell, and I. Lee, "The quest for better clinical word vectors: Ontology based and lexical vector augmentation versus clinical contextual embeddings," *Comput Biol Med*, vol. 134, p. 104433, 2021.

[91] G. Michalopoulos, Y. Wang, H. Kaka, H. Chen, and A. Wong, "Umlsbert: Clinical domain knowledge augmentation of contextual embeddings using the unified medical language system metathesaurus," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021, pp. 1744–1753.

[92] E. Hsu and K. Roberts, "Llm-ie: a python package for biomedical generative information extraction with large language models," *JAMIA Open*, vol. 8, no. 2, p. ooaf012, 03 2025. [Online]. Available: https://doi.org/10.1093/jamiaopen/ooaf012

[93] B. Tang, Y. Wu, M. Jiang, Y. Chen, J. C. Denny, and H. Xu, "A hybrid system for temporal information extraction from clinical text," *Journal of the American Medical Informatics Association*, vol. 20, no. 5, pp. 828–835, 2013.

[94] J. J. Saiz and B. Altuna, "End-to-end temporal relation extraction in the clinical domain." in *Text2Story@ ECIR*, 2023, pp. 13–23.

[95] A. Yadav and D. K. Vishwakarma, "A weighted text representation framework for sentiment analysis of medical drug reviews," in *2020 IEEE sixth international conference on multimedia big data (BigMM)*. IEEE, 2020, pp. 326–332.

[96] C. Colón-Ruiz and I. Segura-Bedmar, "Comparing deep learning architectures for sentiment analysis on drug reviews," *Journal of Biomedical Informatics*, vol. 110, p. 103539, 2020.

[97] F. Gräßer, S. Kallumadi, H. Malberg, and S. Zaunseder, "Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning," in *Proceedings of the 2018 international conference on digital health*, 2018, pp. 121–125.

[98] Y. Mercadier, J. Azé, and S. Bringay, "Divide to better classify," in *Artificial Intelligence in Medicine: 18th International Conference on Artificial Intelligence in Medicine, AIME 2020, Minneapolis, MN, USA, August 25–28, 2020, Proceedings 18*. Springer, 2020, pp. 89–99.

[99] T. Bemila, I. Kadam, A. Sidana, and S. Zemse, "An approach to sentimental analysis of drug reviews using rnn-bilstm model," in *Proceedings of the 3rd international conference on advances in science & technology (ICAST)*, 2020.

[100] Z. Min, "Drugs reviews sentiment analysis using weakly supervised model," in *2019 IEEE international conference on artificial intelligence and computer applications (ICAICA)*. IEEE, 2019, pp. 332–336.

[101] J. Camacho-Collados and M. T. Pilehvar, "On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis," in *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics, 2018.

[102] V. Novotnỳ, E. F. Ayetiran, M. Štefánik, and P. Sojka, "Text classification with word embedding regularization and soft similarity measure," *arXiv preprint arXiv:2003.05019*, 2020.

[103] H. Ennajari, N. Bouguila, and J. Bentahar, "Knowledge-enhanced spherical representation learning for text classification," in *Proceedings of the 2022 SIAM international conference on data mining (SDM)*. SIAM, 2022, pp. 639–647.

[104] Y. Meng, J. Huang, G. Wang, C. Zhang, H. Zhuang, L. Kaplan, and J. Han, "Spherical text embedding," *Advances in neural information processing systems*, vol. 32, 2019.

[105] Y. Lin, Y. Meng, X. Sun, Q. Han, K. Kuang, J. Li, and F. Wu, "Bertgcn: Transductive text classification by combining gcn and bert," *arXiv preprint arXiv:2105.05727*, 2021.

[106] L. Huang, D. Ma, S. Li, X. Zhang, and H. Wang, "Text level graph neural network for text classification," *arXiv preprint arXiv:1910.02356*, 2019.

[107] D. Kim, D. Seo, S. Cho, and P. Kang, "Multi-co-training for document classification using various document representations: Tf–idf, lda, and doc2vec," *Information sciences*, vol. 477, pp. 15–29, 2019.

[108] S. Robertson, "Understanding inverse document frequency: on theoretical arguments for idf," *Journal of documentation*, vol. 60, no. 5, pp. 503–520, 2004.

[109] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[110] B. Sowmya, K. Srinivasa *et al.*, "Large scale multi-label text classification of a hierarchical dataset using rocchio algorithm," in *2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*. IEEE, 2016, pp. 291–296.

[111] T. Ganegedara, *Natural Language Processing with TensorFlow: Teach Language to Machines Using Python's Deep Learning Library*. Birmingham, UK: Packt Publishing Ltd., 2018.

[112] R. Johnson and T. Zhang, "Supervised and semi-supervised text categorization using lstm for region embeddings," in *Proceedings of The 33rd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. F. Balcan and K. Q. Weinberger, Eds., vol. 48. New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 526–534. [Online]. Available: https://proceedings.mlr.press/v48/johnson16.html

[113] S. Berndorfer and A. Henriksson, "Automated diagnosis coding with combined text representations," in *Informatics for Health: Connected Citizen-Led Wellness and Population Health*. IOS Press, 2017, pp. 201–205.

[114] F. K. Khattak, S. Jeblee, C. Pou-Prom, M. Abdalla, C. Meaney, and F. Rudzicz, "A survey of word embeddings for clinical text," *Journal of Biomedical Informatics*, vol. 100, p. 100057, 2019.

[115] P. López-Úbeda, M. C. Díaz-Galiano, T. Martín-Noguerol, A. Luna, L. A. Ureña-López, and M. T. Martin-Valdivia, "Automatic medical protocol classification using machine learning approaches," *Computer Methods and Programs in Biomedicine*, vol. 200, p. 105939, 2021.

[116] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, pp. 391–407, 1990.

[117] Y. Mercadier, "Classification automatique de textes par réseaux de neurones profonds: application au domaine de la santé," Ph.D. dissertation, Université Montpellier, 2020.

[118] M. Arguello Casteleiro, G. Demetriou, W. Read, M. J. Fernandez Prieto, N. Maroto, D. Maseda Fernandez, G. Nenadic, J. Klein, J. Keane, and R. Stevens, "Deep learning meets ontologies: experiments to anchor the cardiovascular disease ontology in the biomedical literature," *Journal of biomedical semantics*, vol. 9, pp. 1–24, 2018.

[119] T. Vu, D. Q. Nguyen, and A. Nguyen, "A label attention model for icd coding from clinical text," in *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 2021, pp. 3335–3341.

[120] C. C. Aggarwal, *Machine learning for text*. Springer, 2018.

[121] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*. PMLR, 2014, pp. 1188–1196.

[122] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[123] E. Batbaatar and K. H. Ryu, "Ontology-based healthcare named entity recognition from twitter messages using a recurrent neural network approach," *International journal of environmental research and public health*, vol. 16, no. 19, p. 3628, 2019.

[124] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of naacL-HLT*, vol. 1, no. 2. Minneapolis, Minnesota, 2019.

[125] I. Beltagy, K. Lo, and A. Cohan, "Scibert: A pretrained language model for scientific text," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3615–3620.

[126] Y. Peng, S. Yan, and Z. Lu, "Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets," in *Proceedings of the 18th BioNLP Workshop and Shared Task*, 2019, pp. 58–65.

[127] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "Biobert: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.

[128] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, M. Walker, H. Ji, and A. Stent, Eds. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 2227–2237.

[129] W. Yin and H. Schütze, "Learning word meta-embeddings," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 1351–1360.

[130] S. M. Meystre, G. K. Savova, K. C. Kipper-Schuler, and J. F. Hurdle, "Extracting information from textual documents in the electronic health record: a review of recent research," *Yearbook of medical informatics*, vol. 17, no. 01, pp. 128–144, 2008.

[131] S. Sikorskiy, O. Metsker, A. Yakovlev, and S. Kovalchuk, "Machine learning based text mining in electronic health records: cardiovascular patient cases," in *Computational Science–ICCS 2018: 18th International Conference, Wuxi, China, June 11–13, 2018 Proceedings, Part III 18*. Springer, 2018, pp. 818–824.

[132] W. Sun, Z. Cai, Y. Li, F. Liu, S. Fang, and G. Wang, "Data processing and text mining technologies on electronic medical records: a review," *Journal of healthcare engineering*, vol. 2018, no. 1, p. 4302425, 2018.

[133] A. T. McCray, J. L. Sponsler, B. Brylawski, and A. C. Browne, "The role of lexical knowledge in biomedical text understanding," in *proceedings of the annual symposium on computer application in medical care*, 1987, p. 103.

[134] H. D. Tolentino, M. D. Matters, W. Walop, B. Law, W. Tong, F. Liu, P. Fontelo, K. Kohl, and D. C. Payne, "A umls-based spell checker for natural language processing in vaccine safety," *BMC medical informatics and decision making*, vol. 7, pp. 1–13, 2007.

[135] D. B. Aronow, F. Fangfang, and W. B. Croft, "Ad hoc classification of radiology reports," *Journal of the American Medical Informatics Association*, vol. 6, no. 5, pp. 393–411, 1999.

[136] L. Zhou and G. Hripcsak, "Temporal reasoning with medical data—a review with emphasis on medical natural language processing," *Journal of biomedical informatics*, vol. 40, no. 2, pp. 183–202, 2007.

[137] L. Zhou, C. Friedman, S. Parsons, and G. Hripcsak, "System architecture for temporal information extraction, representation and reasoning in clinical narrative reports," in *AMIA Annual Symposium Proceedings*, vol. 2005, 2005, p. 869.

[138] A. Rajkomar, E. Loreaux, Y. Liu, J. Kemp, B. Li, M.-J. Chen, Y. Zhang, A. Mohiuddin, and J. Gottweis, "Deciphering clinical abbreviations with a privacy protecting machine learning system," *Nature Communications*, vol. 13, no. 1, p. 7456, 2022.

[139] J. Rogers and A. L. Rector, "Terminological systems: bridging the generation gap," in *Proceedings of the AMIA Annual Fall Symposium*, 1997, p. 610.

[140] G. Héja, G. Surján, and P. Varga, "Ontological analysis of snomed ct," in *BMC medical informatics and decision making*, vol. 8. Springer, 2008, pp. 1–5.

[141] P. Kestel, P. Kügler, C. Zirngibl, B. Schleich, and S. Wartzack, "Ontology-based approach for the provision of simulation knowledge acquired by data and text mining processes," *Advanced Engineering Informatics*, vol. 39, pp. 292–305, 2019.

[142] B.-M. Hsu, "Comparison of supervised classification models on textual data," *Mathematics*, vol. 8, no. 5, p. 851, 2020.

[143] O. Bodenreider, "The unified medical language system (umls): integrating biomedical terminology," *Nucleic acids research*, vol. 32, no. suppl_1, pp. D267–D270, 2004.

[144] P. Bernus, J. Blazewicz, G. Schmidt, and M. Shaw, *International handbooks on information systems*. Springer, 2008. [Online]. Available: https://www.springer.com/series/3795

[145] A. Bandrowski, R. Brinkman, M. Brochhausen, M. H. Brush, B. Bug, M. C. Chibucos, K. Clancy, M. Courtot, D. Derom, M. Dumontier *et al.*, "The ontology for biomedical investigations," *PloS one*, vol. 11, no. 4, p. e0154556, 2016.

[146] M. Skreta, A. Arbabi, J. Wang, E. Drysdale, J. Kelly, D. Singh, and M. Brudno, "Automatically disambiguating medical acronyms with ontology-aware deep learning," *Nature communications*, vol. 12, no. 1, p. 5319, 2021.

[147] I. Harrow, R. Balakrishnan, E. Jimenez-Ruiz, S. Jupp, J. Lomax, J. Reed, M. Romacker, C. Senger, A. Splendiani, J. Wilson *et al.*, "Ontology mapping for semantically enabled applications," *Drug discovery today*, vol. 24, no. 10, pp. 2068–2075, 2019.

[148] G. Deshpande, Q. Motger, C. Palomares, I. Kamra, K. Biesialska, X. Franch, G. Ruhe, and J. Ho, "Requirements dependency extraction by integrating active learning with ontology-based retrieval," in *2020 IEEE 28th international requirements engineering conference (RE)*. IEEE, 2020, pp. 78–89.

[149] M. Chang, G. D'Aniello, M. Gaeta, F. Orciuoli, D. Sampson, and C. Simonelli, "Building ontology-driven tutoring models for intelligent tutoring systems using data mining," *IEEE Access*, vol. 8, pp. 48 151–48 162, 2020.

[150] J. Jovanović and E. Bagheri, "Semantic annotation in biomedicine: the current landscape," *Journal of biomedical semantics*, vol. 8, pp. 1–18, 2017.

[151] Z. Lin, D. Yang, H. Jiang, and H. Yin, "Learning patient similarity via heterogeneous medical knowledge graph embedding," *IAENG International Journal of Computer Science*, vol. 48, no. 4, pp. 868–877, 2021.

[152] N. Nath, S.-H. Lee, M. D. McDonnell, and I. Lee, "The quest for better clinical word vectors: Ontology based and lexical vector augmentation versus clinical contextual embeddings," *Computers in Biology and Medicine*, vol. 134, p. 104433, 2021.

[153] W. Sun, A. Rumshisky, and z. Uzuner, "Annotating temporal information in clinical narratives," *Journal of Biomedical Informatics*, vol. 46, no. Supplement, p. S5–S12, 2013.

[154] A. Stubbs, "Mae and mai: lightweight annotation and adjudication tools," in *Proceedings of the 5th Linguistic Annotation Workshop*. Association for Computational Linguistics, 2011, pp. 129–133.

[155] Ö. Uzuner, B. R. South, S. Shen, and S. L. Duvall, "2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text," *J Am Med Informatics Assoc*, vol. 18, no. 5, pp. 552–556, 2011.

[156] S. Henry, K. Buchan, M. Filannino, . Stubbs, and O. Uzuner, "2018 N2C2 Shared Task on Adverse Drug Events and Medication Extraction in Electronic Health Records," *Journal of the American Medical Informatics Association*, vol. 27, no. 1, pp. 3–12, 2020.

[157] H. Tiaiba, L. Sabri, A. Chibani, and K. Okba, "Machine learning for drug efficiency prediction," in *International Conference on Wireless Mobile Communication and Healthcare*. Springer Nature Switzerland, 2022, pp. 322–337.

[158] I. Turc, M.-W. Chang, K. Lee, and K. Toutanova, "Well-read students learn better: On the importance of pre-training compact models," *arXiv preprint arXiv:1908.08962*, 2019.

[159] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[160] O. Mechi, M. Mehri, R. Ingold *et al.*, "Text line segmentation in historical document images using an adaptive u-net architecture," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2019, pp. 369–374.

[161] Y. Lecun, L. Bottou, Y. Bengio *et al.*, "Gradient-based learning applied to document recognition," *Proceedings Of The IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[162] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances In Neural Information Processing Systems*, vol. 25, 2012.

[163] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," *arXiv preprint arXiv:1603.07285*, 2016.

[164] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1520–1528.

[165] S. Mahalingam and S. Pandraju, "Unsupervised convolutional filter learning for covid-19 classification," *Revue D'intelligence Artificielle*, vol. 35, no. 5, 2021.

[166] A. Rakin, I. Majumder, and M. Akib, "A gan-based federated learning architecture for data augmentation of medical images," 2022.

[167] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[168] Analytics Vidhya, "An introduction to separable convolutions," https://www.analyticsvidhya.com/blog/2021/11/an-introduction-to-separable-convolutions/, accessed: 2022-05-11.

[169] A. G. Howard, M. Zhu, B. Chen *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[170] Q. Li, J. Ning, J. Yuan, and L. Xiao, "A depthwise separable dense convolutional network with convolution block attention module for covid-19 diagnosis on ct scans," *Computers In Biology And Medicine*, vol. 137, p. 104837, 2021.

[171] M. Z. Asghar, F. R. Albogamy, M. S. Al-Rakhami *et al.*, "Facial mask detection using depthwise separable convolutional neural network model during covid-19 pandemic," *Frontiers In Public Health*, vol. 10, p. 855254, 2022.

[172] W. Lin, Y. Ding, H. Wei *et al.*, "Ldsconv: Learned depthwise separable convolutions by group pruning," *Sensors*, vol. 20, no. 15, p. 4349, 2020.

[173] L. Dang, P. Pang, and J. Lee, "Depth-wise separable convolution neural network with residual connection for hyperspectral image classification," *Remote Sensing*, vol. 12, no. 20, p. 3408, 2020.

[174] C. Fran, "Deep learning with depth wise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2017.

[175] S. Dey, R. Bhattacharya, S. Malakar *et al.*, "Covidconvlstm: A fuzzy ensemble model for covid-19 detection from chest x-rays," *Expert Systems With Applications*, vol. 206, p. 117812, 2022.

[176] A. Rayan, A. Alaerjan, S. Alanazi *et al.*, "Utilizing cnn-lstm techniques for the enhancement of medical systems," *Alexandria Engineering Journal*, vol. 72, pp. 323–338, 2023.

[177] X. Shi, Z. Chen, H. Wang *et al.*, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[178] J. Donahue, L. Anne Hendricks, S. Guadarrama *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.

[179] Y. Seo, M. Defferrard, P. Vandergheynst *et al.*, "Structured sequence modeling with graph convolutional recurrent networks," in *Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13-16, 2018, Proceedings, Part I*, vol. 25. Springer, 2018, pp. 362–373.

[180] S. Liu, W. Ma, R. Moore, V. Ganesan, and S. Nelson, "Rxnorm: Prescription for electronic drug information exchange," *IT Prof*, vol. 7, no. 5, pp. 17–23, 2005.

[181] T. Kang, S. Zhang, and Y. Tang, "EliIE: An open-source information extraction system for clinical trial eligibility criteria," *J Am Med Informatics Assoc*, vol. 24, no. 6, pp. 1062–1071, 2017.

[182] J. Kim, Y. Choe, and K. Mueller, "Extracting clinical relations in electronic health records using enriched parse trees," *Procedia Comput Sci*, vol. 53, no. 1, pp. 274–283, 2015.

[183] X. Yang, J. Bian, W. R. Hogan, and Y. Wu, "Clinical concept extraction using transformers," *J Am Med Informatics Assoc*, vol. 27, no. 12, pp. 1935–1942, 2020.

[184] H. Tiaiba, L. Sabri, and O. Kazar, "Intertwining ontologies with deep learning for events and temporal expressions extraction from narrative medical texts," in *2025 7th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*. IEEE, 2025, pp. 1–6.

[185] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, vol. 25, 2012, pp. 1097–1105.

[186] J. F. Allen, "Towards a general theory of action and time," *Artificial Intelligence*, vol. 23, no. 2, pp. 123–154, 1984.

[187] C. Claramunt and B. Jiang, "A representation of relationships in temporal spaces," in *GIS and Geocomputation*. CRC Press, 2000, pp. 53–66.

[188] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," *Advances in Neural Information Processing Systems*, vol. 13, pp. 1137–1143, 2000.

[189] J. D'Souza and V. Ng, "Annotating inter-sentence temporal relations in clinical notes," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA), 2014, pp. 2758–2765.

[190] Y.-C. Chang, H.-J. Dai, J.-C. Wu, and J.-C. Chen, "Tempting system: A hybrid method of rule and machine learning for temporal relation extraction in patient discharge summaries," *Journal Of Biomedical Informatics*, vol. 46, pp. S54–S60, 2013.

[191] C. Cherry, X. Zhu, C. Martin, and B. De Bruijn, "À la recherche du temps perdu: Extracting temporal relations from medical text in the 2012 i2b2 nlp challenge," *Journal of the American Medical Informatics Association*, vol. 20, no. 5, pp. 843–848, 2013.

[192] C. Lin, D. Dligach, T. A. Miller *et al.*, "Multilayered temporal modeling for the clinical domain," *Journal of the American Medical Informatics Association*, vol. 23, no. 2, pp. 387–395, 2016.

[193] J. D'Souza and V. Ng, "Classifying temporal relations in clinical data: A hybrid, knowledge-rich approach," *Journal of Biomedical Informatics*, vol. 46, no. S29–S39, 2013.

[194] A. Nikfarjam, E. Emadzadeh, and G. Gonzalez, "Towards generating a patient's timeline: Extracting temporal relationships from clinical notes," *Journal of Biomedical Informatics*, vol. 46, no. S40–S47, 2013.

[195] C.-Y. Cheng, P. Anick, P. Hong *et al.*, "Temporal relation discovery between events and temporal expressions identified in clinical narrative," *Journal of Biomedical Informatics*, vol. 46, no. S48–S53, 2013.

[196] Y. Lee, Hsien-Ju and, M. Jiang, J. Xu *et al.*, "Identifying direct temporal relations between time and events from clinical notes," *BMC Medical Informatics And Decision Making*, vol. 18, no. 2, pp. 23–34, 2018.

[197] C. Cheng and J. C. Weiss, "Typed markers and context for clinical temporal relation extraction," *Proceedings of Machine Learning Research*, vol. 219, pp. 94–109, Aug 2023.

[198] H. Tiaiba, L. Sabri, and O. Kazar, "Advanced prediction of events and temporal expressions in medical text using the jena api: Integrating ontologies and deep learning," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 33, no. 1, 2025.

[199] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 7370–7377.

[200] H. Zhu and P. Koniusz, "Simple spectral graph convolution," in *International conference on learning representations*, 2021.

[201] F. Wu, A. Souza, T. Zhang, C. Fifty, T. Yu, and K. Weinberger, "Simplifying graph convolutional networks," in *International conference on machine learning*. PMLR, 2019, pp. 6861–6871.

[202] H. Tiaiba, L. Sabri, A. Chibani, and O. Kazar, "Artificial neural network for multi-label medical text classification," in *2022 2nd International Conference on New Technologies of Information and Communication (NTIC)*. IEEE, 2022, pp. 1–7.

[203] J.-C. Na and W. Y. M. Kyaing, "Sentiment analysis of user-generated content on drug review websites," *Journal of Information Science Theory and Practice*, vol. 3, no. 1, pp. 6–23, 2015.

[204] M. E. Basiri, M. Abdar, M. A. Cifci, S. Nemati, and U. R. Acharya, "A novel method for sentiment classification of drug reviews using fusion of deep and machine learning techniques," *Knowledge-Based Systems*, vol. 198, p. 105949, 2020.

[205] S. Vijayaraghavan and D. Basu, "Sentiment analysis in drug reviews using supervised machine learning algorithms," *arXiv preprint arXiv:2003.11643*, 2020.

[206] Y. Mercadier, "Classification automatique de textes par réseaux de neurones profonds: application au domaine de la santé," Ph.D. dissertation, Université Montpellier, 2020.

LIST OF PUBLICATIONS

Journals' papers

- TIAIBA, HAFIDA; SABRI, LYAZID; and KAZAR, OKBA (2025) "Advanced prediction of events and temporal expressions in medical text using the Jena API: integrating ontologies and deep learning," Turkish Journal of Electrical Engineering and Computer Sciences: Vol. 33: No. 1, Article 6. https://doi.org/10.55730/1300-0632.4114 Available at: https://journals.tubitak.gov.tr/elektrik/vol33/iss1/6
- Hafida Tiaiba, Lyazid Sabri, and Okba Kazar, "Ontologies and Machine Learning Models to Enhance Health Informatics: A Survey, Challenges and Future Directions," IAENG International Journal of Applied Mathematics, vol. 55, no. 3, pp475-499, 2025 Available at: https://www.iaeng.org/IJAM/issues_v55/issue_3/IJAM_55_3_03.pdf

International Conferences

- Tiaiba, H., Sabri, L., Chibani, A., & Kazar, O. (2022, December). Artificial Neural Network for Multi-label Medical Text Classification. In 2022 2nd International Conference on New Technologies of Information and Communication (NTIC) (pp. 1-7). IEEE.
- Tiaiba, H., Sabri, L., Chibani, A., & Kazar, O. (2022, November). Machine Learning for Drug Efficiency Prediction. In International Conference on Wireless Mobile Communication and Healthcare (pp. 322-337). Cham: Springer Nature Switzerland.
- H. Tiaiba, L. Sabri and O. Kazar, "Intertwining Ontologies with Deep Learning for Events and Temporal Expressions Extraction from Narrative Medical Texts," 2025 7th International Conference on Pattern Analysis and Intelligent Systems (PAIS), Laghouat, Algeria, 2025, pp. 1-6, doi: 10.1109/PAIS66004.2025.11126522.