Aristotle University of Thessaloniki

Faculty of Sciences

School of Informatics

MSc Data and Web Science

Investigating Entity Linking in Greek Electronic Health Records: Leveraging Hierarchical Structures and Bi-Encoder Architectures

Alkis Kipouros

ID: 124

Supervisor: Grigorios Tsoumakas

March 26, 2024

Inscription

To those who strive for a healthier tomorrow through the relentless pursuit of knowledge.

?

Thanks to

I would like to express my gratitude to my supervisor, Grigorios Tsoumakas whose guidance, support, and expertise have been invaluable throughout this research journey. I am also profoundly grateful for this opportunity I was given to work on such an interesting project, which provided a platform for valuable academic exploration.

My appreciation extends to my colleagues Axilleas Toumpas and Eleonora Stoikopoulou who collaborated with me during the initial phases of setting up this project. Their contributions were instrumental in laying the foundation for this research endeavor. Additionally, my sincere thanks are extended to the medical professionals Athanasios Samaras and Alexandra Bekiaridou who generously shared their expertise and data, and whose dedicated efforts in the data annotations were what made this research possible.

Lastly, heartfelt thanks to my family and friends for their unwavering support, understanding, and encouragement, providing the foundation for this academic endeavor.

Abstract

Entity linking in the context of Greek medical texts presents unique challenges due to the unstructured nature of Electronic Health Records and the scarcity of resources in the Greek language. This study proposes a novel methodology embracing two distinct paradigms for EL tasks. The first paradigm adopts a one-stage EL approach, employing a standalone hierarchical classifier that leverages the natural hierarchy of the ICD-10. The second paradigm explores a two-stage EL strategy involving candidate concept generation and reranking. The proposed approach integrates the robustness of a bi-encoder architecture for candidate generation with a hierarchical classifier for reranking. This dynamic combination aims to enhance the effectiveness of EL in the intricate landscape of Greek medical texts. The study systematically evaluates both architectures independently and jointly. The hierarchical classifier undergoes meticulous assessment, comparing alternative configurations to identify optimal settings for Greek EHR entity linking. Joint evaluations with the bi-encoder for candidate generation and independent assessments of the bi-encoder's ability to retrieve correct entities further contribute to a comprehensive understanding of their strengths and limitations. This research not only addresses the challenges of EL in Greek medical texts but also contributes insights into evolving strategies for EL tasks, shedding light on optimal configurations and novel paradigms for tackling intricacies within the domain of healthcare.

Keywords: Natural Language Processing, Information Extraction, Medical concept normalization, Medical entity linking, Electronic Health Records (EHRs), Greek Medical Tests, ICD-10

Περίληψη

Η συχέτιση οντοτήτων (entity linking, EL) στο πλαίσιο των Ελληνικών ιατρικών κειμένων αντιμετωπίζει μοναδικές προκλήσεις λόγω της μη δομημένης φύσης των Ηλεκτρωνικών Ιατρικών Εγγράφων και της έλλειψης πόρων στην ελληνική γλώσσα. Η συγκεκριμένη μελέτη προτείνει μια νέα μεθοδολογία που υιοθετεί δύο παραδείγματα για τη συσχέτιση οντωτήτων. Το πρώτο παράδειγμα υιοθετεί μια προσέγγιση ενός σταδίου, χρησιμοποιώντας ένα αυτόνομο ιεραρχικό ταξινομητή που εκμεταλλέυεται τη φυσική ιεραρχία του ICD-10. Το δεύτερο παράδειγμα εξετάζει μια στρατιγική δύο σταδίων που περιλαμβάνει τη δημιουργία υποψηφίων εννοιών και την αναδιάταξή τους (candidate generation and reranking). Η προτεινόμενη προσέγγιση ενσωματώνει την ανθεκτικότητα μιας αρχιτεκτονικής με διπλό κωδικοποιητή (bi-encoder) για τη δημιουργία υποψήφιων έννοιων με έναν ιεραρχικό ταξινομητή για την αναδιάταξη. Αυτός ο δυναμικός συνδυασμός στοχεύει στη βελτίωση της αποτελεσματικότητας του EL στο περίπλοκο τοπίο των ελληνικών ιατρικών κειμένων. Η μελέτη αξιολογεί συστηματικά και ανεξάρτητα τις δύο αρχιτεκτονικές. Ο ιεραρχικός ταξινομητής υφίσταται λεπτομερή αξιολόγηση, συγκρίνοντας εναλλακτικές διαμορφώσεις για να εντοπίσει τις βέλτιστες ρυθμίσεις για τον συσχετισμό των ηλεκτρονικών ιατρικών κειμένων. Συνδυαστικές αξιολογήσεις με τον bi-encoder για τη δημιουργία υποψηφίων και ανεξάρτητες εκτιμήσεις της δυνατότητας του να ανακτήσει ακριβείς οντότητες προσθέτουν στη κατανόηση των δυνατοτήτων και των περιορισμών του. Η μελέτη αυτή αντιμετωπίζει όχι μόνο τις προκλήσεις EL σε ελληνικά ιατρικά κείμενα, αλλά παρέχει επίσης εισηγήσεις στην εξέλιξη των στρατηγικών EL, φωτίζοντας τις βέλτιστες διαμορφώσεις και τα καινοτόμα παραδείγματα για την αντιμετώπιση των ιδιαιτεροτήτων στον τομέα της υγείας.

Λέξεις κλειδιά: Επεξεργασία Φυσικής Γλώσσας, Εξαγωγή Πληροφοριών, Κανονικοποίηση Ιατρικών Έννοιων, Σύνδεση Ιατρικών Οντοτήτων, Ηλεκτρονικά Υγειονομικά Έγγραφα (EHRs), Ελληνικά Ιατρικά Κείμενα, ICD-10

Contents

1 Introduction **10**

2 Background **14**

2.1 Navigating Healthcare Information: NER and NEN Basics 14

2.2 Evolution of Approaches in Information Extraction from Clinical Texts 15

2.3 Deep Learning Approaches in NLP 16

2.4 Electronic Health Records 18

2.5 Annotation Tools and Medical Knowledge Bases 19

3 Related Work **22**

3.1 Biomedical Corpora: An Analysis of Prior Efforts 22

3.1.1 English-Language Corpora 22

3.1.2 International Corpora 23

3.1.3 Conclusions 26

3.2 Entity Linking Systems 26

3.3 Recent Developments in the Biomedical Information Extraction Field 34

4 Idea and Approach **43**

4.1 Data Exploration 43

4.1.1 The structure of ICD-10 43

4.1.2 Data Analysis 45

4.2 Methodology 48

4.2.1 Hierarchical Classifier 50

4.2.2 Bi-Encoder 54

5 Implementation and Experimentation **59**

5.1 Creation of the Dataset 59

5.1.1 Document De-identification 59

5.1.2 The Annotation process 60

5.1.3 Shaping the Final Dataset Form 61

5.2 Implementation 62
5.2.1 Data Preprocessing 62
5.2.2 Hierarchical Classifier 63
5.2.3 ICD-10 Bi-Encoder 66
5.3 Experimentation Setup 67
5.3.1 Dataset 67
5.3.2 Models 68
5.3.3 Evaluation 68

6 Results and Discussion **71**
6.1 Evaluation of the Hierarchical model as a standalone classifier 71
6.2 Evaluation of the Bi-Encoder 73
6.3 Evaluation with document-level labels 76
6.3.1 Analysis of Unpredicted Labels 78

7 Conclusions and Future Work **83**
7.1 Conclusions 83
7.1.1 Key Findings and Implications 84
7.1.2 Limitations 86
7.2 Future Work 87

List of Figures

4.1 ICD-10 Structure Example 45
4.2 Distribution of the 50 most frequent ICD-10 codes 46
4.3 Distribution of the 50 most frequent ICD-10 codes divided by chapter 46
4.4 Distribution of ICD-10 Chapters 47
4.5 Hierarchical Classifier Architecture 51
4.6 Mention-Entity Bi-Encoder Architecture 54
4.7 Two-staged EL combining the bi-encoder and the hierarchical cls 57

5.1 Structure of a typical discharge document 60
5.2 Anootation prosses on the doccano environment 61
5.3 Tabular dataset form 61

6.1 Relationship Between the Number of k and Candidate Set Accuracy in Bi-Encoder Evaluation 76
6.2 Distribution of the number of document-level labels 77
6.3 Hierarchical Cls Label Misclassification Distribution 79
6.4 Bi-encoder Label Misclassification Distribution 80

List of Tables

3.1 Summary of Biomedical Corpora 27
3.2 Entity Linking Aspects and their Approaches 35
3.3 Summary of Biomedical EL Approaches 42

4.1 ICD-10 Chapter Descriptions 44
4.2 Statistical Summary of ICD-10 Code Occurrences 48

5.1 Wordpiece Examples 63
5.2 Mention and context examples 63
5.3 Learning Rate Results for the context-aware Hierarchical classifier 64
5.4 Common hyperparameters shared by all classifier models. 65
5.5 Model architecture and training parameters for different classifier configurations 66
5.6 Contents of train, test and validation sets 68
5.7 Model architecture and training parameters for the Bi-encoder model 69

6.1 Hierarchical Classifier Evaluation Results 72
6.2 Evaluation of two-stage EL for various numbers of candidates 74
6.3 Bi Encoder's ability to retrieve the correct label within the top-k candidates . 75
6.4 Model Evaluation for document-level labels 78
6.5 Hierarchical cls' most frequent mispredicted labels and corresponding mentions 79
6.6 Bi-encoder's most frequent mispredicted labels and corresponding mentions . 81
6.7 Bi-Encoder Analysis Summary 82

List of Algorithms

Chapter 1

Introduction

In recent years, the landscape of natural language processing (NLP) has evolved significantly, driven by the rapid advancements in large language models (LLMs) and transformer-based technologies. This evolution has extended to healthcare, particularly in the field of clinical NLP and health data processing, unlocking unprecedented opportunities for extracting valuable insights from medical texts.

The backbone of all patient data is found in Electronic Health Records (EHRs), rich in both volume and information. These records offer a comprehensive snapshot of an individual's medical history, encompassing current diagnoses, personal and family history, procedures, lab tests, medications, symptoms, and personalized treatment plans. They stand as a crucial reservoir of empirical data for biomedical research, providing an extensive platform to explore various aspects of health, from diseases like Alzheimer's and cardiovascular conditions to evaluating associated risk factors and monitoring adverse drug reactions. Expressing this enormity and unprecedented detail, EHRs are a valuable source of insights. Dealing with EHRs presents a unique challenge, because in addition to the information of the structured data of the lab test and demographics they include unstructured and diverse data in the form of natural language text. Unlike conventional datasets, EHRs embody a wealth of unstructured and diverse medical narratives, introducing complexity through rich clinical language, varied sentence structures, and the inclusion of freeform text. Recent advancements in Natural Language Processing (NLP), particularly with Large Language Models (LLMs) and transformer-based models, represent a breakthrough in addressing the intricate challenges posed by processing EHRs.

The collaboration of medicine and artificial intelligence (AI) shows great potential, ready to change how medical information is extracted. This significant change is crucial, especially when we consider the wealth of information stored in the narrative structure of EHRs. These advancements find various applications, beginning with Automated Information Extraction. Automated Information Extraction involves leveraging AI technologies, such as NLP and

machine learning algorithms, to automatically extract relevant information from unstructured clinical narratives within EHRs. This includes diagnoses, treatments, patient history, and other pertinent details. The traditional manual labor associated with data extraction from EHRs involves clinicians and healthcare professionals manually reviewing patient records, identifying key information, and entering it into structured databases or electronic systems. The implementation of Automated Information Extraction mitigates the need for extensive manual labor by automating the extraction process. NLP algorithms are trained to understand the intricacies of medical language, recognize named entities (such as diseases, medications, and procedures), and identify relationships between these entities. As a result, AI systems can efficiently process large volumes of unstructured clinical text, extracting valuable information accurately and at a much faster pace than manual methods, saving time and human resources when medical professionals are themselves, in shortage.

Despite recent progress, this task still demands the expertise of qualified medical professionals. Yet, the integration of AI significantly reduces this dependency, thanks to its robust natural language processing capabilities. The impact can extend further with Clinical Decision Support Systems, fortified by the analyzed EHR data, that emerge as a tool to provide insights, flag potential issues, and furnish evidence-based recommendations. Predictive analytics, fueled by historical patient data, allows for the anticipation of disease progression, identification of potential complications, and optimization of treatment plans, paving the way for proactive and personalized patient care.

Unlocking the full potential of this information wealth marks a significant advancement in biomedical research. It opens avenues for expanding the quantity and diversity of scientifically usable data, gaining new insights into diseases, identifying risk factors, and enhancing healthcare services. Recognizing this untapped potential, recent initiatives aim to overcome the language-bound barrier of EHRs. The primary objective is evidently to transform this extensive information into a resource that is both accessible and actionable for researchers.

Information extraction is typically divided into two steps. Named Entity Recognition (NER) and Named Entity Normalization (NEN), also to be referred as Entity Linking (EL). NER is an NLP technique focused on identifying and classifying named entities (such as persons, organizations, locations, dates, and other specific terms) within a body of text. The goal is to extract structured information from unstructured text, providing a foundation for understanding the key entities mentioned in the text and their relationships. In the context of medical texts, NER helps identify entities like symptoms, diagnoses, drug names, and other relevant information, contributing to the overall process of information extraction and

analysis.

This work primarily focuses on the task of medical Entity Linking. Entity Linking is an important mechanism of information extraction as it gives meaning to chunks of natural text. At its core, entity linking seeks to establish connections between explicit mentions of entities within a document and their corresponding representations in a predefined knowledge base or ontology. In the context of medical texts found in EHRs, Entity Linking has the important role of structuring and standardizing the information embedded in freeform medical language by associating specific terms or phrases with their canonical representation in a knowledge base. This process is fundamental in unlocking the true potential of medical texts and facilitating downstream applications. Essentially, Entity Linking works like a bridge connecting the specific terms used in medical texts to their standardized versions in the larger medical knowledge base, and the standardization ensures consistency in understanding and interpreting medical information across different documents and systems. Moreover, by enabling cross-domain insights, it opens doors to a deeper understanding of diseases and risk factors.

Medical texts pose unique challenges for entity linking due to their rich intricacies. The diverse expressions used for procedures in clinical content add complexity, requiring robust systems for comprehensive recognition. Given the pivotal role of procedures in healthcare, it's essential to develop advanced systems capable of identifying a broad spectrum of treatment options, diagnostic procedures, and therapeutic techniques associated with patient care.

Furthermore, EHRs introduce linguistic phenomena that make entity linking more challenging. The incorporation of specialized terminology and the dynamic and variable nature of sentence structures ranging from intricate and complex structures to incomplete formats with abbreviated terms and medical professional jargon make the task all more challenging. The contextual nuances of terms, the abundance of synonyms, and the prevalence of multi-word expressions add layers of intricacy, thereby intensifying the complexity of extracting medical information from EHRs.

It is noteworthy to explore the advancements made in non-English medical texts. While numerous studies on medical information extraction have been conducted in English, research for other languages, with Spanish being the second most studied, is limited to almost non-existent. This scarcity of resources, particularly the absence of datasets for research, poses a significant challenge. Therefore, making a global effort to address these linguistic limitations becomes paramount. This ensures that advancements in medical entity linking are not confined to a specific linguistic context but can be extended to diverse languages,

thereby broadening the impact of such technologies on a global scale.

In line with these goals, this work is driven by two overarching objectives:

1. Collaboration with Medical Professionals for Dataset Creation
    - Collaborate with medical professionals to manually annotate Electronic Health Records for Named Entity Recognition and Entity Linking tasks in the Greek Language.
    - Aim to create a comprehensive and accurately annotated dataset that serves as a valuable resource for training and evaluating medical information extraction models.
2. Entity Linking System Development for Greek EHRs Dataset
    - Investigate the capabilities of BERT-based LLMs in encoding Greek medical texts.
    - Develop models for the task of entity linking of the Greek EHRs to ICD-10 codes, and assess their performance in various configurations.

Chapter 2

Background

In the landscape of NLP, the foundational concepts of NER and NEN have undergone a transformative journey, evolving from rule-based approaches to sophisticated deep learning methods. This progression is particularly significant in the context of processing EHRs, where the unstructured medical narratives present distinct challenges. As DL approaches gained prominence in NLP, their application to medical texts became instrumental in extracting and linking entities accurately. Within this context, the advent of annotation tools has played a crucial role in training and fine-tuning NLP models for specific tasks. Additionally, the utilization of medical knowledge bases, such as the SNOMED-CT, UMLS and ICD-10, provides standardized codes for medical entities, facilitating interoperability and semantic understanding. This section delves into the evolution of information extraction and NLP, EHR basics, and the role of annotation tools and knowledge bases in the context of medical entity linking.

2.1 Navigating Healthcare Information: NER and NEN Basics

In the domain of healthcare, the sheer volume of data surpasses human capacity for effective management, necessitating the integration of machine and deep learning approaches. Healthcare data, often presented in EHRs and diverse medical texts, contains a wealth of unstructured information. This complexity arises from the rich clinical language, varied sentence structures, and the inclusion of freeform text. The field that comes forward to help analyze and extract meaningful data from such texts is Natural Language Processing (NLP).

Central to NLP are two critical tasks: Named Entity Recognition (NER) and Named Entity Normalization (NEN). NER is the process of identifying entities within unstructured, natural language text. These entities can encompass a wide range, including person names, organizations, locations, medical codes, time expressions, and more. In the context of healthcare, entities often revolve around vital information like symptoms, diagnoses, drug names, and medical procedures.

Once entities are identified through NER, the subsequent task of NEN comes into play. NEN involves linking these identified entities to standardized concepts within a knowledge base or ontology. This step transforms raw textual data into a structured format, enabling a consistent association between entities and their canonical representations. For example, in a medical text discussing patient symptoms, NER would identify entities like ”fever” or ”headache,” and NEN would link these entities to specific concepts in a medical knowledge base.

NER and NEN play a vital role in healthcare by structuring and standardizing information within the complex landscape of freeform medical language. This normalization ensures a consistent understanding of medical concepts across various textual representations. Particularly in EHRs, where patient information is often recorded in narrative form, these techniques become essential. By adopting NLP techniques, healthcare systems can enhance information extraction, streamline decision-making processes, and unlock the full potential of the wealth of data contained within medical texts.

Named Entity Normalization, or Entity Linking, is in most works defined as two separate steps, following the identification of mention boundaries in the text. The first step is that of candidate generation (CG) and the second step is Entity Ranking (ER).

In the candidate generation step, the goal is to generate a set of potential entities or candidates for each identified mention in the text. This involves creating a pool of possible entities that the mention might refer to. Various methods are employed for candidate generation, ranging from simple lexical matching to more sophisticated approaches involving knowledge bases, gazetteers, or machine learning models. The effectiveness of the candidate generation process significantly impacts the success of the subsequent normalization step.

Following candidate generation, the Entity Ranking (ER) step involves selecting the most appropriate candidate from the generated set for each mention. The task is to rank the candidates based on their relevance to the mention, considering factors such as semantic similarity, context coherence, or other relevant features. Different systems employ diverse strategies for entity ranking, leveraging techniques like rule-based approaches, graph-based methods, machine learning algorithms, and deep learning.

2.2 Evolution of Approaches in Information Extraction from Clinical Texts

The objective of information extraction in the medical field has been approached in three major ways. With rule-based approaches, feature-based approaches and deep learning (DL) approaches. The first systems developed for this task initially adopted rule-based approaches

before transitioning to incorporate feature-based techniques. Eventually, DL models took center stage, demonstrating state-of-the-art performance.

Among the pioneers in entity extraction from medical texts was the Linguistic String Project - Medical Language Processor by Sager et al. [1], utilizing term vocabularies and heuristic rules. Early systems, including Medical Language Extraction and Encoding System (MedLEE) and MetaMap [2, 3], heavily relied on rule-based models. However, these approaches proved fragile, particularly when applied to texts from new subdomains or institutions. Initially, rule-based approaches seamlessly integrated NER and NEN tasks. With evolving methodologies, these tasks began to be performed sequentially. The NER task witnessed the introduction of feature-based models and deep learning solutions, such as Hidden Markov Models [4] and Conditional Random Fields [5] that fall in the first category. For deep learning approaches, a range of models have been seen including Word2Vec and Transformers (GPT and multiple versions of BERT).

The NEN task underwent similar categorizations: rule-based approaches [6], feature-based approaches, and deep learning methods, often in various combinations [7]. In the next section, a detailed exploration of DL approaches for both NER and NEN tasks will be presented.

Recent advancements also showcase systems that integrate NER and NEN tasks, departing from the sequential application. Notable examples include the work of Durrett and Klein [8], employing a CRF model to simultaneously train NER and NEN, and Leaman et al. [9], proposing a semi-Markov Models architecture merging NER and NEN during training and inference. This integrated approach is also evident in literature leveraging deep learning models, where the entity extraction task is not explicitly split into NER and NEN [10, 11].

2.3 Deep Learning Approaches in NLP

Deep learning methods have revolutionized biomedical entity normalization, playing a pivotal role in extracting nuanced information from vast datasets. This section delves into key models—Word2Vec, ELMo, BERT, and GPT—that have significantly contributed to the field. Starting with the simplest of models, Word2Vec has been widely adopted to pre-train word embeddings from large corpora [6]. Word2Vec is a shallow context-independent word embedding model which learns fixed-sized vectors for words based on the context in which they appear. It uses a simple neural network architecture with either a skip-gram or continuous bag-of-words (CBOW) approach [12]. Word2Vec typically does not consider contextual information from neighboring words, and for that reason, it is typically used for tasks where a

simple, context-independent representation of words is sufficient. The advantage it has over other deep-learning models is that it is not very computationally demanding and it works well with smaller datasets.

ELMo (Embeddings from Language Models)[13] is another noteworthy model for word embeddings, ELMo distinguishes itself from Word2Vec by generating context-dependent vectors. These vectors vary based on the context, allowing for multiple embeddings for the same word depending on its meaning. ELMo employs a deep, bi-directional LSTM (Long Short-Term Memory) architecture to capture intricate word representations. However, this context-awareness comes at a computational cost, making it more demanding than the relatively lightweight Word2Vec. The key distinctions between the two models can be summarized into three categories. The primary and most significant difference lies in their level of context dependency. Additionally, they employ different architectures, with Word2Vec utilizing a shallow neural network and ELMo relying on a deep NN architecture (LSTM), contributing to the latter's higher computational load. Lastly, Word2Vec, designed for simpler tasks, requires less data compared to ELMo, which, due to its context dependency, necessitates more data for accurate comprehension.

Two prominent LLMs are currently found in the center of things for NLP tasks, BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer). What stands out immediately is their common foundation—the transformative Transformers architecture. Transformers have become a cornerstone in NLP, reshaping how sequential data is processed. Originating from "Attention is All You Need," [14] transformers depart from conventional sequential models by introducing a self-attention mechanism. This mechanism, central to transformers, allows the model to assign varying importance to different words within a sequence during processing, revolutionizing language understanding and generation. The self-attention mechanism facilitates the simultaneous capture of contextual relationships across the entire sequence—a departure from the limitations of sequential models. Transformers bring efficiency and scalability to the forefront by processing input sequences in parallel, a crucial advantage when dealing with large datasets and complex language structures. Adopting an encoder-decoder architecture, transformers, as seen in NLP tasks like language understanding (as exemplified by BERT) or language generation (as demonstrated by GPT), overcome the inherent challenge of understanding sequential order. The incorporation of positional encodings imparts information about the position of each token in the sequence.

Delving into the exploration of the two notable transformer architectures, BERT and GPT,

BERT [15], specializing in capturing bidirectional context for understanding, has become a staple in biomedical entity extraction literature. Its proven effectiveness in enhancing state-of-the-art practices is attributed to its unique ability to capture contextual nuances from both left and right directions. The subsequent exploration of these models will unravel the distinctive architectures of BERT and GPT. BERT, designed as a bidirectional model, processes input sequences bidirectionally, capturing context comprehensively. On the other hand, GPT [16], functioning as a generative model, employs a unidirectional transformer architecture, processing input sequences from left to right in an autoregressive manner. This architectural distinction sets the stage for their differing training objectives—GPT excels in predicting the next word based on left context, while BERT predicts "masked words," consulting context in both left and right directions. Beyond architecture, the versatility of BERT shines through its ability to be fine-tuned for various NLP tasks, including text classification, named entity recognition, and question answering. In summary, BERT stands out for its proficiency in capturing bidirectional context, making it suitable for diverse NLP tasks and effective for specialized applications through fine-tuning. In contrast, GPT excels in text generation, coherent language understanding, and creative applications, showcasing its aptitude for tasks requiring autoregressive capabilities.

2.4 Electronic Health Records

EHRs serve as digital repositories capturing comprehensive information about a patient's medical journey, including health problems, medications, progress notes, procedures, vital signs, and overall medical history. These electronic versions, maintained by hospitals and medical facilities, contain invaluable data and insights for the medical field.

The sheer volume and complexity of textual data within EHRs underscore the critical need for automated mining. Unlocking insights from this wealth of information becomes key for advancing medical practices, and it is precisely the application of machine learning and deep learning technologies that holds immense promise. Through these technological advancements, healthcare professionals gain access to systems capable of assisting in clinical decision-making, treatment planning, and personalized patient care. This not only empowers medical advancements but also contributes to improving overall patient outcomes.

The unique characteristics of natural language data in EHRs add an extra layer of complexity to their processing. As highlighted by Nesterov and Umerenkov [17], EHRs exhibit distinct phenomena, including rich special terminology, complex sentence structures with missing parts and punctuation, numerous abbreviations, and context-dependent terms often

presented with a multitude of synonyms. Physicians frequently employ shorthand acronyms that may lack widespread recognition and conform poorly to standardized formats [18].

One significant challenge associated with Electronic Health Records is the scarcity of annotated documents. The limited availability of labeled data poses hurdles for the development and training of effective machine-learning models in this domain.

2.5 Annotation Tools and Medical Knowledge Bases

The journey towards training systems for NER and NEN begins with manual annotation of the natural language text. In this process, field experts manually identify entities within the text and link them to corresponding concepts from a knowledge base. This is usually performed with the help of software called annotation tools. Some examples used in previous literature, like in the works of Syed et al. [19] and Chen et al. [20] include brat rapid annotation tool and Doccano.

The *brat rapid annotation tool (brat)* [21] is a web-based, open-source annotation tool designed for intuitive visualization and editing of text annotations. It supports various annotation primitives, including typed text spans, binary relations, n-ary associations, and attributes/meta-knowledge. Notable features include entity normalization support, embedded visualizations, and discontinuous text annotations. brat's visualization was originally designed for event extraction but extended for diverse annotation tasks. The tool offers mouse-based editing gestures for easy annotation creation and connection. Developed in collaboration with experienced annotators, brat implements features like automatic validation against semantic constraints. Brat is freely available with full source code, accessible through its homepage and code repository.

Doccano stands as an open-source data labeling tool catering to machine learning practitioners. Its collaborative capabilities and language-agnostic support make it a versatile choice. Designed for easy installation and a user-friendly experience, the platform boasts an intuitive interface. Its flexibility shines through in accommodating various annotation tasks, including NER and NEN, sentiment analysis, translation, intent detection, image classification, captioning, and more. As an open-source solution, doccano fosters a collaborative environment for annotators and researchers, enabling efficient work on diverse text and image annotation projects. Its popularity within the annotation community stems from user-friendly features and robust support for multiple tasks, facilitating collaborative efforts in crafting annotated datasets for training machine learning models.

For a comprehensive annotation process, an essential component is a knowledge base.

In clinical text annotation, this base should consist of concepts representing medical terms, procedures, and relevant entities. Previous literature showcases the utilization of knowledge bases such as the Unified Medical Language System (UMLS) [22], [23], SNOMED-CT [24] [25], and scarcely, ICD-10 [26].

The Unified Medical Language System (UMLS) [27], developed by the National Library of Medicine, integrates biomedical vocabularies, terminologies, classifications, and coding systems. It comprises a Semantic Network illustrating relationships between concepts and a Metathesaurus consolidating concepts from diverse source vocabularies. Serving as a comprehensive resource for interoperability in biomedical information systems, including electronic health records, the UMLS promotes seamless communication across healthcare platforms. Accessible through a licensing process, individuals can utilize the UMLS Terminology Services (UTS) account, which encompasses three knowledge sources: the Metathesaurus, Semantic Network, and SPECIALIST Lexicon and Lexical Tools. These sources cover terms and codes from various vocabularies, broad semantic categories, and a syntactic lexicon equipped with tools for normalizing strings and generating lexical variants.

Systematized Nomenclature of Medicine - Clinical Terms (SNOMED-CT) is globally recognized as an International Standard for clinical terminology. SNOMED CT serves as a versatile terminology with the capability to cross-map to various international terminologies, classifications, and code systems. Mapping involves associating specific concepts or terms in one system with counterparts in another system, sharing similar meanings. The primary goal of mapping is to establish a link between different international terminologies, classifications, and code systems, yielding several advantages. These advantages encompass data reuse, enabling clinical data based on SNOMED CT to be repurposed for reporting statistical and management data using other terminologies. Mapping ensures the retention of data value during migrations to newer database formats and schemas, minimizing the need for redundant data entry and mitigating associated risks of increased costs and errors. Additionally, it fosters interoperability among international terminologies, classifications, and code systems. SNOMED CT, beyond its English language primary release, also offers a Spanish translation directly managed by SNOMED International.

ICD-10 (International Classification of Diseases, 10th Edition) is a coding system for diseases, conditions, and related health issues widely used for billing and statistical purposes in healthcare. It is vital for international standardization of disease coding, aiding in data exchange and analysis. ICD-10-CM is a clinical modification used for diagnosis coding, while ICD-10-PCS is used for procedure coding. The International Classification of Diseases (ICD), overseen

by the World Health Organization (WHO), serves as a globally accepted medical classification system, integral to epidemiology, health management, and clinical diagnostics. Originating in 1983, the tenth revision (ICD-10) was endorsed in 1990, becoming operational in WHO Member States by 1994. This comprehensive system facilitates the tracking of over 55,000 codes, enabling the monitoring of new diagnoses and procedures—an extensive upgrade from the 17,000 codes in ICD-9. Its adoption has been widespread, with many countries incorporating adaptations of their own.

In summary, UMLS acts as a unifying framework that integrates various biomedical vocabularies. SNOMED CT is a detailed clinical terminology system emphasizing relationships between concepts. ICD-10 is a coding system specifically focused on diseases and health conditions for billing and statistical purposes. While UMLS provides a broader integration, SNOMED-CT and ICD-10 serve more specific roles in the representation and coding of clinical information.

Annotation tools and knowledge bases serve as foundational pillars in biomedical information extraction. In the next chapter, the focus shifts to the application of the previously discussed concepts in innovative and state-of-the-art systems, delving into the exploration of corpora that play a pivotal role in shaping the landscape of biomedical research. Following the exploration of corpora, attention will be directed toward entity linking systems and the challenges and advancements in connecting entities within the intricate web of biomedical information.

Chapter 3

Related Work

3.1 Biomedical Corpora: An Analysis of Prior Efforts

Dedicated efforts in medical entity annotation have resulted in a multitude of corpora, each contributing to the advancement of biomedical text mining and information extraction. This section provides an overview of several notable corpora.

3.1.1 English-Language Corpora

The *NCBI Disease Corpus* [28] which is a resource developed by the National Center for Biotechnology Information (NCBI) for advancing biomedical text mining and information extraction in the domain of diseases. The corpus comprises a collection of manually annotated PubMed abstracts. The NCBI Disease Corpus was one of the early endeavors to provide a comprehensive and annotated dataset specifically focusing on diseases. Its structured annotation schema and emphasis on disease-related information laid a foundation for subsequent projects. The annotations include information about disease mentions and their corresponding concepts. As an earlier work, it has served as an inspiration and a benchmark for future research. The NCBI Disease Corpus has served as a benchmark for evaluating and developing models for disease recognition, normalization, and relationship extraction. Researchers often use it as a reference in the development and assessment of clinical NLP model architectures.

A large clinical entity corpus developed in 2018 by Patel et al. [22], includes 5,160 documents annotated for 11 entity types across 40 domains. The documents underwent an initial classification into different domains and work types while emphasizing on capturing diverse medical domains, work types, and physical expertise. The annotations were built upon the UMLS semantic groups, while 11 clinical entity types were defined, covering problems. procedures, anatomical structures, and more. Furthermore, some relationships were explored, including those between anatomical structures, measurement values, and modifiers. The annotators involved were of microbiology and biochemistry backgrounds, while inter-

annotator agreement was achieved through discussion and refinement reaching a Kappa score of 96.89% for entity and relationship annotation. In the end, they achieved 443,328 annotated concepts, with top categories including Problem, Procedure, Medication, and Anatomical Structure. Upon that corpus, a CRF system was utilized for entity recognition.

MedMentions [29] is a biomedical corpus consisting of a total of 4,392 abstracts from PubMed. The UMLS Metathesaurus, encompassing concepts from over 200 source ontologies, was selected as the foundation. Professional annotators used the GATE tool to manually annotate UMLS entity mentions from the abstracts. An evaluation involving two biologists achieved a 97.3% agreement, estimating the precision of the annotation. The UMLS, while expansive, contains concepts that may not be ideal for specialized document retrieval. Filtering strategies, such as creating the ST21pv subset, were proposed to address this challenge.

3.1.2 International Corpora

Beyond English-language corpora, several international initiatives focusing on languages like Spanish, French, Russian and Portuguese yielded important advancements in the field of biomedical NLP:

The NEREL-BIO corpus [30] is a biomedical annotation scheme and corpus for PubMed abstracts in Russian and English. It addresses the lack of richly annotated datasets in Russian biomedical NLP. Extending the general domain dataset NEREL, it introduces domain-specific entity types and covers nested named entities that cross boundaries, posing a detection challenge. With annotations for 700+ Russian and 100+ English abstracts, NEREL-BIO serves as a benchmark for cross-domain and cross-language transfer. It is freely available, allowing experiments with transformer-based models. To create NEREL-BIO, 766 Russian and 105 English abstracts from WMT-2020 Biomedical Translation Task were annotated, utilizing mBERT-based NER models and BRAT annotation. Seventeen specialized biomedical entity types and 20 from NEREL were included, linked to UMLS. NEREL-BIO handles diverse entity mentions and introduces changes in annotation principles. Inter-annotator agreement, measured by Krippendorff's alpha, indicates good reliability.

MERLOT [31] and *APcNER* [32] are corpora both created on clinical narratives in French. The first one is annotated for linguistic, semantic, and structural information, to support clinical information extraction. The corpus was annotated by six annotators using a comprehensive annotation scheme that covers 21 entities, 11 attributes, and 37 relations. An automatic tool was used to produce entity and attribute pre-annotations. The performance of the pre-annotation tool for entities reached an F-measure of 0.814 when sufficient train-

ing data was available. To ensure annotation consistency across the corpus, harmonization tools were devised to automatically identify annotation differences that needed resolution, thereby improving overall corpus quality. The corpus comprises 500 documents, totaling 148,476 tokens, and is annotated with 44,740 entities and 26,478 relations. The average inter-annotator agreement is reported as 0.793 F-measure for entities and 0.789 for relations. For the APcNER corpus, 147 documents were randomly sampled from the AP-HP de-identification dataset. Document types included discharge summaries, letters from physicians, operating reports, and additional examination reports. The annotation was based on UMLS semantic types for 5 medical entities: Drug names, Signs or symptoms, Diseases or disorders, Diagnostic procedures or lab tests, and Therapeutic procedures. A pre-annotation was performed using a terminology-based annotator while the final corpus was annotated by a medical resident (IL), and quality assessed by a medical doctor (NG) on 10 randomly selected documents. A simple conditional random field (CRF) model was applied to detect annotation inconsistencies or case errors, and manual corrections were performed based on the results. Brat annotation tool was used for both these corpora.

SemClinBr [23] is a corpus made upon Portuguese clinical texts. Two datasets from Brazilian hospitals, spanning 2002 to 2018, were used, including diverse medical specialties and narrative types. A total of 1,000 clinical narratives were randomly chosen, covering various medical specialties and document types. The annotation schema, based on UMLS semantic types, included entities such as signs, symptoms, diseases, and procedures. Relationship types were also defined. The corpus demonstrated good Inter-Annotator Agreement (IAA), with 0.71 for strict and 0.92 for relaxed concept annotation. Relations achieved an IAA of 0.86. An extensive analysis in the errors was also conducted, spotting that errors were mainly due to word span differences and high granularity in semantic types. Ambiguities in UMLS assignments and occasional confusion in concept boundaries were also noted. SemClinBr was successfully applied in negation detection and clinical named entity recognition, yielding very promising results.

The *CT-EBM-SP* corpus [33] is a collection of 1,200 texts about clinical trials written in Spanish, including 700 from EudraCT, 500 abstracts, and 52 related to COVID-19. The corpus consists of formal, scientific literature abstracts and clinical trial protocols. Abstracts are longer and more complex, while protocols contain simpler language in specific sections. The authors identified errors, misspellings, and mistranslations, especially in EudraCT texts. The annotation process involved pre-annotation using a hybrid NER pipeline, annotating four entity types (pathologies, anatomical entities, biochemical/pharmacological substances, and

procedures) based on UMLS semantic groups. The inter-annotator agreement was measured using the F-measure. Three NER frameworks were tested on the corpus: SequenceLabeler, Flair, and BERT. The results indicate the suitability of the CT-EBM-SP corpus for NER tasks in the medical domain.

The CodiEsp corpus [26] comprises 1,000 clinical case reports in Spanish across various medical specialties. Professional clinical coders manually annotated all documents with codes form the Spanish version of ICD-10 (diagnostic and procedural). The annotation process adhered to official clinical coding guidelines, and clinical codes were linked to supporting textual evidence fragments. The annotations underwent refinement for quality, achieving high pairwise agreement percentages. The corpus statistics reveal a total of 18,435 annotations. The corpus contains 3,427 unique ICD-10 codes, with 2,557 diagnostic codes and 870 procedural codes. The generation of additional resources included a machine-translated version of the corpus, CodiEsp-abstracts, PubMed machine-translation, and the CodiEsp Silver Standard, offering a larger dataset for evaluation. These resources aim to facilitate comparisons, support non-Spanish speaking participants, and explore machine-translated corpora in clinical coding systems.

DisTEMIST [34] is a collection of 1000 clinical case reports in Spanish, manually annotated and linked to SNOMED-CT terms. Clinical cases from different databases were gathered, preprocessed, and manually classified. The DisTEMIST project also delved into the creation of a Multilingual Silver Standard corpus, incorporating six languages: English, Portuguese, Catalan, Italian, French and Romanian.Neural machine translation was used for translating text files and annotations individually. Manual error analysis identified issues such as synonyms, gendered words, and word inflection while potential solutions were identified like creating morphological variant lists and using more intricate lookup systems.

MedProcNER Gold Standard Corpus [35] is a collection of 1,000 clinical case reports in Spanish from various medical specialties. Manually annotated and normalized with SNOMED CT mentions of procedures. In addition, it comprises more than 300,000 concepts in 19 different hierarchies, consisting of 234,674 lexical entries, with 130,219 main terms and 130,219 unique codes. Documents were obtained from SciELO, an electronic library with publications from Latin America, South Africa, and Spain. The 1,000 documents were manually selected by a practicing oncologist for relevance, richness, and variety. It was annotated and standardized by two clinical experts from a Spanish tertiary hospital and later reviewed by a third physician, reaching a total IAA score of 81.2%. A Multilingual Silver Standard corpus was also released, derived from the Spanish Gold Standard in 9 languages: English, Catalan, Italian,

French, Portuguese, Romanian, Czech, Dutch and Swedish.

3.1.3 Conclusions

Examining these diverse biomedical corpora collectively reveals common themes and variations, providing insights into the evolution of efforts in medical Entity Linking. Given the study's nature, greater emphasis has been placed on the aspect of international corpora. Beyond the established English resources, a range of languages has emerged, including Russian, French, Portuguese, and various works in the Spanish language. The development of Silver Standard corpora has expanded to cover a promising variety of languages. Regarding the annotation schemes and knowledge bases used in these works, the predominant choice is the utilization of UMLS semantic types, followed by the SNOMED-CT KB. Notably, only one of the works incorporates the ICD-10, recognized as the least commonly employed. The corpora exhibit diverse domain coverage, with some focusing on specific domains and others adopting a broader coverage. In conclusion, the comparison of these biomedical corpora reveals the collective efforts and advancements in medical entity linking. While each corpus has its unique contributions and characteristics, the shared emphasis on standardized annotation, diverse domain coverage, and the integration of advanced tools underscores the collaborative nature of research in this field. Table 3.1 is provided to better summarize the key characteristics of the corpora.

3.2 Entity Linking Systems

As discussed, entity linking is an important part of information extraction, connecting textual mentions to their corresponding entities in a knowledge base. Over the years, a plethora of Entity Linking Systems have been developed, each leveraging various methodologies to tackle the complexities inherent in this task. This section provides a comprehensive overview of existing Entity Linking Systems, exploring their underlying mechanisms, strengths, and limitations. By delving into the landscape of these systems, valuable insights are gained from the evolution of Entity Linking technologies and the diverse strategies employed to achieve accurate and contextually aware linking. From rule-based approaches to advanced machine learning models, the spectrum of Entity Linking Systems reflects the continuous pursuit of refining techniques that bridge the semantic gap between natural language text and knowledge base.

A survey that examined advancements in neural entity linking models to the year 2022 [36], showcased their superior accuracy compared to classical methods. The authors present

Corpus	Language	Knowledge Base	Consists of
NCBI Disease Corpus	English	UMLS	793 PubMed abstracts
Clinical Entity Corpus	English	UMLS	5,160 clinical documents
MedMentions	English	UMLS	4,392 abstracts
NEREL-BIO	Russian, English	UMLS	766 Russian and 105 English abstracts
MERLOT	French	UMLS	500 documents
APcNER	French	UMLS and SNOMED	147 documents
SemClinBr	Portuguese	UMLS	1,000 clinical narratives
CT-EBM-SP	Spanish	UMLS	1,200 abstracts and texts about clinical trials
CodiEsp Corpus	Spanish	ICD-10	1,000 clinical case reports
DisTEMIST Gold Standard	Spanish	SNOMED-CT	1000 clinical case reports
DisTEMIST Silver Standard	English, Portuguese, Catalan, Italian, French and Romanian	SNOMED-CT	Machine-translated versions of the Gold Standard corpus files
MedProcNER Gold Standard	Spanish	SNOMED-CT	1,000 clinical case reports
MedProcNER Silver Standard	English, Catalan, Italian, French, Portuguese, Romanian, Czech, Dutch and Swedish	SNOMED-CT	Machine-translated versions of the Gold Standard corpus files

Table 3.1: Summary of Biomedical Corpora

a generic neural entity linking architecture applicable to a broad spectrum of neural EL systems. The architecture encompasses crucial components such as candidate generation, entity ranking, mention and entity encoding. Various modifications to this architecture are categorized into four common directions: (1) joint entity mention detection and linking models, (2) global entity linking models, (3) domain-independent approaches (including zero-shot and distant supervision methods), and (4) cross-lingual techniques.

In the first category, the survey explores the simultaneous solution of entity mention detection and disambiguation, making the task more challenging but potentially improving the overall pipeline's quality. The joint models in recent neural entity linking systems necessitate the production of mention candidates. Various strategies are employed, such as enumerating all spans in a sentence, treating each token n-gram as a mention candidate, or considering various possible spans. Some models leverage multitask learning, proposing stack-based bidirectional LSTM networks with a shift-reduce mechanism and attention for entity recognition. A notable approach in this direction is the end-to-end method by Broscheit [37], formulating entity linking as a sequence labeling problem. This approach assigns each token in the text an entity link or a NIL class, employing a sequence tagger based on pre-trained BERT. Poerner et al. [38]. propose E-BERT-MLM, repurposing the masked language model (MLM) objective for candidate selection. Cao et al. [39] introduce a generative approach based on BART, performing sequence-to-sequence autoregressive generation of text markup with information about mention spans and links to entities. While these joint models face increased complexity, their natural mutual dependency between mention detection and disambiguation steps can lead to improved overall performance.

In the realm of entity disambiguation (ED), contextual information can be categorized into local and global contexts. The survey delves into global context architectures, where the disambiguation decision for one entity is influenced by decisions made for other entities in a context. Unlike local approaches that consider each mention independently, global methods involve semantic consistency across multiple entities. Various strategies are employed in global disambiguation, with most methods facing the challenge of combinatorial complexity and the difficulty of assigning consistency scores to entities. One typical approach involves generating a graph encompassing candidate entities and applying graph algorithms, such as random walk algorithms or graph neural networks. Recent advancements include neural recurrent random walk networks, dynamic graph convolution architectures, and attention mechanisms that consider subgraphs of target mentions. Despite optimizations, globally considering coherence scores among candidates of all mentions simultaneously can be com-

putationally slow and susceptible to erroneous coherence among wrong entities. To address this, some studies frame the global ED problem as a sequential decision task. Reinforcement learning is employed to train policy networks for sequential entity selection, leveraging knowledge from previously linked entities. Iterative prediction models, attention mechanisms, and features from previously linked entities are also explored to enhance sequential approaches. Additionally, studies introduce feed-forward neural networks and document-wide context to implicitly capture coherence, avoiding explicit design of an entity coherence component.

Achieving domain independence is a crucial aspect of Entity Linking (EL) systems, given the scarcity of annotated resources across various domains. Traditional approaches utilized unsupervised and semi-supervised models, relying on surface-matching heuristics and binary multi-instance learning. Recent studies, particularly in the context of distant learning and zero-shot methods, aim to address domain independence challenges. Le and Titov [40] propose distant learning techniques that leverage unlabeled documents, relying on weak supervision from surface matching heuristics. The EL task is framed as binary multi-instance learning, distinguishing between positive entities and random negatives. While promising, these approaches require a Knowledge Graph (KG) describing entity relations or mention-entity priors computed from entity hyperlink statistics. Zero-shot methods focus on adapting EL systems to new domains where only textual descriptions of entities are available. These techniques train EL systems on domains with rich labeled data and apply them to new domains with minimal data. Candidate generation in zero-shot approaches involves pre-computing representations of entity descriptions, allowing for efficient similarity calculations during inference. Embeddings and BERT-based bi-encoders are employed for candidate generation, with recent studies using BERT-based cross-encoders for joint encoding of mentions and entities. For entity ranking in zero-shot methods, embedding-based approaches and BERT-based cross-encoders exhibit competitive performance. The cross-attention mechanism, especially in Logeswaran et al. [41] and Wu et al. [42]'s studies, prove effective in leveraging semantic information from both the context and entity descriptions. Nie et al. [43] incorporate cross-attention with recurrent architectures, and B. and Zhang [44] address the limitation of standard BERT in handling long contexts. Evaluation of zero-shot systems requires datasets from diverse domains, and heavy neural architectures pre-trained on open corpora significantly enhance performance. Further exploration of unsupervised pre-training on source and target data is deemed beneficial. Closing the performance gap between fast representation-based bi-encoders and computationally intensive cross-encoders remains an open question. Additionally, developing better approaches for utilizing unlabeled data could

be a promising avenue for future research.

In response to the shortage of labeled data for Entity Linking (EL) outside of English, Cross-Lingual Entity Linking methods (XEL) aim to overcome this limitation by leveraging supervision from resource-rich languages, often relying on Wikipedia as a valuable source of cross-lingual information. The use of inter-language links in Wikipedia proves essential for mapping entities seamlessly across diverse languages. However, the challenge arises from the lack of mappings between mention strings and entities in resource-poor languages. To address this challenge, various methods have been proposed, including the utilization of inter-language links, translation dictionaries, translation models, neural string matching models, and even tapping into online search engine results. Fu et al. [45] critique the exclusive reliance on Wikipedia and introduce a candidate generation method that incorporates online search engines, demonstrating superior recall. In dealing with the scarcity of annotated examples, several approaches leverage cross-lingual data. Pan et al. [46] employ Abstract Meaning Representation (AMR) statistics and mention context for ranking, incorporating pseudo-labeling for AMR tagger training. Tsai and Roth [47] opt for training monolingual embeddings for words and entities, projecting them into the English space and subsequently averaging context embeddings for ranking. Sil et al. [48] propose a zero-shot transfer method, enhancing existing approaches with embedding projection learning, a CNN context encoder, and trainable re-weighting. Upadhyay et al. [49] extend zero-shot approaches by integrating global context and typing information, yielding notable performance improvements when using mention-entity priors. While many techniques rely on pre-trained multilingual embeddings for entity ranking, their effectiveness in settings with prior probabilities contrasts with a significant drop in performance in realistic zero-shot scenarios. The recent success of zero-shot multilingual transfer using large pre-trained language models inspires the exploration of potent multilingual self-supervised models. Botha et al. [50] capitalize on the zero-shot monolingual architecture, constructing a massively multilingual EL model for over 100 languages. Their system excels, particularly in zero-shot and few-shot settings, underscoring the advantages of training on extensive multilingual data.

Notably, the majority of studies still depend on external knowledge for candidate generation, while mention encoders have shifted from convolutional and recurrent models to self-attention architectures, incorporating pre-trained contextual language models like BERT. A recent surge in methods addresses domain adaptation in a zero-shot fashion, showcasing the ability to adapt a model trained in one domain to another without annotated data in the target domain, relying solely on entity descriptions. Cross-encoder architectures emerge as

superior, with studies indicating their effectiveness compared to models with separate mention and entity encoders. Global context is widely utilized, though a few recent studies focus exclusively on local entity linking. Among jointly performing mention detection and entity disambiguation models, the entity-enhanced BERT model (E-BERT) and an autoregressive model based on BART stand out. For local models in disambiguation, notable solutions include those by Shahbazi et al. [51] leveraging entity-aware ELMo (E-ELMo) and Wu et al. [42] based on a BERT bi-/cross-encoder, especially effective in zero-shot settings. Yamada et al. [52] report consistently superior results attributed to a masked entity prediction mechanism for entity embedding and the use of a pre-trained model based on BERT with a multi-step global scoring function.

Moving on to specific case studies, three key questions about the entity linking task were set to be answered in by the aforementioned work of Broscheit [37]: (a) Can BERT learn all entity linking steps jointly? (referring to mention detection, candidate generation and entity disambiguation as the three steps) (b) How much entity knowledge is already present in pretrained BERT and (c) Does additional entity knowledge improve BERT's performance in downstream tasks? To address these questions, the authors propose an extreme simplification of the entity linking setup, treating it as a per-token classification over the entire entity vocabulary. The entity vocabulary is based on the 700K most frequent entities in English Wikipedia. The study finds that this simplified model improves entity representations over plain BERT and outperforms architectures that optimize the tasks separately. However, it comes second to the state-of-the-art method that jointly performs mention detection and entity disambiguation. The paper also explores the usefulness of entity-aware token representations in various benchmarks, including GLUE, SQUAD V2, SWAG, and EN-DE WMT14. Surprisingly, the study reveals that most benchmarks do not benefit from additional entity knowledge, except for the RTE task in GLUE, which sees a 2% improvement. The contributions of the paper include being the first to study the joint learning of mention detection, candidate generation, and entity disambiguation in a fully neural model. It also introduces the concept of modeling entity linking as a token classification task and highlights the lack of tasks that evaluate additional entity knowledge in pretrained language models.

Zero-shot entity linking (ZEL) presents a challenging scenario wherein a system is tasked with linking entities not encountered during training, thus mirroring a more realistic use case. B. and Zhang [44] addressed the challenges associated with zero-shot entity linking, emphasizing the limitations of existing BERT-based models. The primary drawback identified was the fixed input length, which could lead to the oversight of crucial dispersed information

across documents. To overcome these limitations, the authors introduced the Bidirectional Multi-Paragraph Reading (Bi-MPR) model. This model innovatively treats the mention context as a query, facilitating matching with multiple paragraphs from entity description documents. To capture essential information, the Bi-MPR model employs an entity-mention attention mechanism. Notably, it incorporates both forward and backward matching steps to mitigate the challenge of insufficient mention context. An inter-paragraph attention mechanism enhances semantic understanding during the forward matching step. In the training process, the authors introduced a novel masking strategy called "Whole Entity Masking" (WEM) and proposed a Unidirectional Multi-Paragraph Reading (UniMPR) model. Notably, an extra pre-training stage is incorporated before fine-tuning, a strategy tailored for each target domain. The UniMPR model addresses the limited length of the baseline model's entity description by matching multiple paragraphs in the candidate entity document with the mention context. A unique aspect of WEM involves selectively masking words within randomly selected entity names, compelling the model to predict entities by understanding their contexts. Despite the improvements introduced by Uni-MPR in reading more paragraphs in entity documents, it has notable limitations, especially when key information resides in paragraphs beyond the mention context. To address this, the Bi-MPR model is proposed, aiming to leverage more textual information in mention documents. Unlike Uni-MPR, Bi-MPR avoids matching each mention-entity paragraph pair, instead emphasizing bidirectional matching between mention and entity documents to enhance entity linking performance.

For evaluation, the study utilized a zero-shot entity linking dataset, with the top-64 candidates for each mention retrieved through the BM25 algorithm. The experimental results demonstrated the superiority of the proposed models (Uni-MPR and Bi-MPR) over the baseline model in terms of average accuracy. The efficacy of the WEM strategy was evident in its ability to learn superior representations compared to vanilla random masking. Particularly in domains with longer documents, Uni-MPR and Bi-MPR showcased substantial improvements. The feasibility and effectiveness of extending input length and adopting multi-paragraph reading were further validated, underscoring the models' enhanced accuracy in handling the zero-shot entity linking task.

NeSLET [53], a neuro-symbolic approach, was proposed for Zero-Shot entity linking using multi-task learning. A neuro-symbolic approach refers to the integration of neural network-based models and symbolic reasoning methods in artificial intelligence and machine learning systems. It combines the strengths of neural networks, which excel at learning from data, with symbolic reasoning, which leverages explicit knowledge representation and

logical inference. NeSLET, or Neuro-symbolic Entity Linking using Entities Type, represents a novel approach in the field of entity linking, as it employs a multi-task learning approach, simultaneously training on both the primary task of entity linking and an auxiliary task of hierarchical entity type prediction. It predicts not only the entity linked to a mention but also the hierarchical type of that entity in a logical hierarchy, providing a more nuanced understanding of entities. This MTL strategy allows the model to leverage information from both tasks, enhancing its overall performance. The paper claims to be the first to demonstrate that incorporating the hierarchical structure of entity types leads to improved entity linking performance when compared to a flat treatment. This highlights the importance of considering the relationships and hierarchies within entity types for more accurate linking. Experiments on four benchmark datasets demonstrate that NeSLET outperforms state-of-the-art baselines (BLINK and GENRE) in multiple low-data regimes. Overall, the paper aims to overcome the challenge of high resource requirements in training ZEL models by introducing a novel approach that combines symbolic information with existing models, achieving competitive performance with significantly less training data.

The incorporation of *Bi-encoders* has marked a significant leap forward in the realm of entity linking, while noteworthy being the introduction of an innovative Candidate Generation model by Partalidou et al. [54]. This model, leveraging a BERT-based bi-encoder, employs two independent BERT transformers for encoding mention-entity pairs, a design choice that not only facilitates real-time inference but also enables efficient caching of candidate representations. The structured input for the bi-encoder is meticulously crafted by encoding mention context and entity descriptions using special tokens and entity type information. The exploration of various pooling functions further enhances the model's capability to encode mentions and entities effectively in the same dense space, achieving state-of-the-art results. Incorporating entity types, identified through spaCy, into the model's structured input adds valuable constraints to potential entity links. The experimentation with different retrieval methods, including k-nearest neighbors classification and similarity measures like cosine similarity, dot product, and Euclidean distance, contributes to the construction of robust candidate sets. Notably, the proposed method demonstrates an impressive accuracy of 84.28% on the top-50 candidates in the Zeshel dataset, surpassing the previous 82.06% on the top-64 candidates. While the inclusion of entity type side-information marginally enhances the performance of pooling functions, the overall promising results on both seen and unseen entity datasets position this method as a valuable complement to existing entity linking approaches.

A diverse array of innovative technologies has emerged to tackle the distinctive challenges associated with entity linking, bringing substantial advancements to the field through the integration of neural network technologies. Several models have attempted to simultaneously address Named Entity Recognition (NER) and Named Entity Normalization (NEN) tasks, employing varied methods such as enumerating all spans in a sentence, treating each token n-gram as a mention candidate, and utilizing sequence labeling techniques. While a single BERT model has demonstrated satisfactory performance by jointly learning all entity linking tasks, models that separate NER from NEN have reported superior results. Global entity models, incorporating contextual information from other entities, have been introduced, employing graph-based methods for entity representation. Techniques such as neural network recurrent random walk networks, dynamic graph convolution, and attention mechanisms have been leveraged in these endeavors. To address challenges related to erroneous coherence among incorrect entities, strategies such as reinforcement learning and iterative prediction models have been employed.

Domain independence has emerged as a critical consideration, particularly due to limited annotated resources. Adapting entity linking models to low-resource domains stands out as a significant challenge. This task is further compounded by the scarcity of resources in entity linking systems designed for low-resource languages, underscoring the continual need for research and development efforts. Notably, the introduction of solutions like zero-shot entity linking has gained prominence. These solutions involve training entity linking systems on domains or languages with rich labeled data and subsequently applying them to new domains. Bi-encoders have shown promising results, particularly in the candidate generation step, while cross-encoders have been effectively employed for entity reranking. The exploration and integration of these techniques mark pivotal strides in addressing the complexities associated with domain adaptation and resource constraints in the realm of entity linking. A summary of the discussed techniques can be found in table 3.2.

3.3 Recent Developments in the Biomedical Information Extraction Field

Navigating the landscape of recent developments in the biomedical information extraction field, it's imperative to recognize the persistent challenges stemming from domain dependence and limited resources. These challenges are particularly pronounced in the relatively underexplored realm of biomedical entity linking and, moreover, in the domain of biomedical Natural Language Processing (NLP) for low-resource languages. This section meticulously examines notable strides made in the field, unveiling cutting-edge techniques that have proven

Entity Linking Aspects	**Strategies**
Handling NER and NEN tasks	Simultaneously with single BERT model vs. separate NER and NEN models
Global entity models leveraging contextual information	Neural recurrent random walk networks, dynamic graph convolution, attention mechanisms
Addressing challenges of coherence among incorrect entities	Reinforcement learning, iterative prediction models
Domain dependence and Low-Resource Challenges	**Strategies and Solutions**
Adapting EL models to low-resource domains	Pre-training on domains with high-resource domains
Entity linking for low-resource languages	Supervision from resource-rich languages, inter-language links, translation models and dictionaries
Zero-shot entity linking	Bi-encoders for candidate generation, Cross-encoders for entity reranking

Table 3.2: Entity Linking Aspects and their Approaches

effective in the intricate domain of biomedical information extraction.

An early proposed method to achieve high results in the low source field of medical entity linking is through transfer learning by Gligic et al. [18]. In this work, the researchers employed transfer learning by pretraining word embeddings on a large pool of unannotated electronic health records (EHRs) and using them as a foundation for various neural network (NN) architectures. This approach allowed the NN models to learn from the hidden information present in the unannotated EHRs and improve their performance in named entity recognition and extracting relationships between medical terms. Two embeddings versions were created, one with Continuous Bag od Words (CBoW) and one with Continuous Skip-Gram (CSG). Various architectures were explored for the term Classification. The baseline architecture was a Context-Free Feedforward Neural Network (FFN). Upon that, and extension was added to create a context aware FFN, replacing the single-word input with the concatenation of ‘w’ words around the mention token. The final architecture was an RNN. It sequentially read all words in the target window around the target word, and input to the architecture was one word embedding per time step, fully connected to an LSTM layer of 100 units, with the final being fed to a SoftMax function. For the results, no clear superiority was observed between CBOW and CSG in intrinsic evaluation. A context-free FFN was trained to classify

words into target classes or none and both CBOW and CSG show similar F1 scores, with CBOW converging earlier and having more stable performance. For the term classification, The RNN outperforms the other models across all target terms, except for " reason", with its performance rivaling the winner algorithm of the I2B2 2009 challenge.

By 2019, despite BERT's success in various NLP processing tasks, its effectiveness in biomedical and clinical contexts remained underexplored. The creation of EhrBERT [24], a model resulting from the fine-tuning of BioBERT [55] on a substantial dataset of 1.5 million unlabeled electronic health record (EHR) notes brought success in achieving SOTA performance outperforming existing systems like MetaMap and DNorm in terms of F1 scores across different corpora. For the entity normalization task, the methodology treats it as a text classification challenge. The model architecture leverages bidirectional transformers and SoftMax layers for prediction. The parameters are initialized with EhrBERT, and the training objective involves maximizing the log-likelihood of gold annotations. The Adam algorithm is employed for back-propagation during training. The study concludes with EhrBERT achieving higher scores than BioBERT and BERT in almost all datasets, but having similar performance to BioBERT. This underscores the significance of the model domain in influencing its performance, highlighting the crucial role of domain-specific pretraining data in achieving optimal results. Considering the size of the training data, the performance comparison between an EhrBERT model trained on 500k documents and another on 1M documents suggests that the larger dataset exhibited slightly better performance, although this improvement was statistically significant only in one dataset. This implies that opting for a smaller dataset could be advantageous in terms of computational resources and time without substantially sacrificing performance.

Continuing in the trend of BERT advancements in the clinical domain, it was noticed that despite the considerable advancements in leveraging pre-trained models like BERT, BioBERT, and ClinicalBERT [56] for various NLP tasks such as named entity recognition (NER), relation classification (RC), and question answering (QA) across both general and biomedical domains, there was a noticeable gap in research addressing the applicability of these models to biomedical entity normalization, before the work of Ji et al. [57]. In their investigation, an entity normalization architecture was achieved by fine-tuning pre-trained BERT, BioBERT, and ClinicalBERT models. Extensive experiments were conducted to assess the efficacy of these pre-trained models for the entity normalization task, utilizing diverse datasets within the biomedical domain. The methodology followed the two-staged traditional approach of EL, consisting of a candidate generation step and and entity ranking step. For the first

step they generated candidate concepts for each mention using an information retrieval (IR) based method, involving indexing and retrieval of top 10 concepts. For the second step they reranked candidate concepts using BERT/BioBERT/ClinicalBERT models, transforming the task into a sentence-pair classification task. To deal with unlinkable mentions, they used a threshold in the ranking step. The fine-tuning procedure involved constructing input sequences and using a softmax function for ranking scores. The results of this research found that by fine-tuning pre-trained BERT/BioBERT/ClinicalBERT models, they outperformed previous methods and improved biomedical entity normalization accuracy by up to 1.17%. Furthermore, BioBERT and ClinicalBERT models, designed for biomedical domains, outperformed their generic BERT counterparts on specific datasets, emphasizing the relevance of domain-specific models for biomedical entity normalization.

An approach involving semantic type prediction was introduced to deal with the overgeneration of candidate concepts in entity linking, by Vashishth et al. [58]. This work emphasizes that candidate generation is an under-studied component of medical information extraction compared to mention detection and disambiguation. As traditional methods have relied on dictionary lookup and string matching, they achieved high precision with low coverage, new NN methods use entire concept inventories providing complete coverage at the cost of large candidate set sizes. Thus, they proposed *MedType*, a deep learning-based system specifically designed for semantic type prediction. MedType operates in two key steps: MedType Predict predicts the semantic type of a medical entity mention, and MedType Filter refines the candidate set based on the predicted type, producing a filtered set containing concepts aligned with the predicted semantic type. The standard two-step entity linking procedure, comprising candidate generation and disambiguation benefits from semantic type prediction. The introduction of semantic type filtering as an intermediate step effectively curtails candidate overgeneration, therefore enhancing the efficiency of the final disambiguation stage. Challenges associated with semantic types are mitigated by a mapping strategy that condenses the 127 UMLS Metathesaurus types into 24 groups, facilitating multi-label semantic type prediction and filtering. MedType leverages a pre-trained BERT encoder, specifically BioBERT, designed for biomedical corpora, to handle polysemous tokens and capture long-range dependencies. The resulting embeddings are processed through a feed-forward classifier to predict semantic types In evaluations, BioBERT demonstrates consistent improvements in semantic type prediction and filtering, with noteworthy enhancements for NCBI and Bio CDR even in a restricted evaluation setting. However, the most substantial improvements are observed on ShARe and MedMentions datasets, underscoring that the benefits of semantic type filtering

predominantly manifest in the entity linking phase of the information extraction pipeline. An in-depth analysis of MEDTYPE errors on the validation split of the MedMentions dataset reveals an overall proficiency but non-uniform performance across different semantic types. Despite these nuances, semantic type filtering consistently proves to be a valuable enhancement to entity linking performance across various information extraction tools. The observed performance gains, particularly on ShARe and MedMentions datasets, underscore the positive impact of semantic type filtering on refining the entity linking process, emphasizing its significance in biomedical information extraction pipelines.

An approach that deviates from the traditional two-step process was proposed that utilizes a single-step multi-label classification task by Nesterov and Umerenkov [17]. By introducing distant supervision, this study aims to address challenges related to human resource-intensive annotation processes. It is also noteworthy to mention that this study utilizes two EHR datasets in the Russian language, for training and testing. The first one consisting of 2,248,359 visits by 429,478 patients and the second one 1,728,259 visits, involving 694,063 patients. The UMLS was used as a knowledge base, and in particular, the MedDRA module was used as a term translation tool for the Russian language. Weak supervision was employed through a straightforward rule-based model to label both the training and testing datasets. The process involved merging the Electronic Health Record (EHR) dataset with a list of terms from the medical knowledge base (KB). Utilizing a sliding window, all potential candidates were selected and subsequently compared to synonyms of medical entities. Exact matches were identified as positive cases. In this study, the EhrRuBERT model was utilized, pretraining a BERT-Base model with RuBERT weights on a large corpus of electronic health records.The pretraining involved a masked language modeling task with a specific focus on medical domain token embeddings, facilitated by a tokenizer tailored for medical texts. A linear classification layer with 10,000 outputs was added, initialized with weights to ensure a low initial prediction probability for all classes. As a result, it was found that the use of a single transformer model for one-step medical entity extraction from EHRs was excelling in classifying common entities, while challenges arise with certain classes. The researchers establish that 50,000 training examples are sufficient for near-perfect recall, even for challenging classes, and the model generalizes well to diverse cases.

The MedProcNER Task

Venturing into recent developments, it is of great interest to delve into the MedProcNER task at BioASQ 2023, utilizing the previously mentioned Gold Standard MedProcNER corpus [35].

The approaches demonstrated by MedProcNER task participants serve as a notable display of emerging trends in Natural Language Processing (NLP) and information extraction over the recent years, showcasing some of the latest methodologies.

The MedProcNER task has introduced three distinct challenges for participants. The first task, Clinical Procedure Recognition, entails a named entity recognition (NER) objective, where participants are tasked with autonomously identifying references to clinical procedures within a compilation of Spanish clinical case reports. The second task, Clinical Procedure Normalization, delves into entity linking (EL), demanding participants to craft systems capable of assigning SNOMED CT codes to the mentions identified in the preceding sub-task. These tasks collectively aim to advance the field of medical concept recognition and entity linking, providing valuable benchmarks for future applications across various languages. For the EL task, they built a simple baseline model where they assigned to each mention found in the test set the same code it had in the training set.

The EL subtask saw a diverse array of systems, incorporating both supervised and unsupervised methods. Participants predominantly utilized approaches centered on semantic search and textual similarity for the normalization task.

Team Vicomtech, leading the competition, utilized SapBERT [59] representations along with cross-encoder architectures, as detailed in their paper [60]. Their approach involved Transformer-based Semantic Search, utilizing SapBERT-XLMR-large to embed entity words and SNOMED CT descriptions. The model, pretrained with UMLS database using XLM-RoBERTa-large as the base, demonstrated the advantage of incorporating multilingual clinical terminology knowledge into a pretrained language model. The embedding dimension of 1024 proved sufficient without truncation, with the [CLS] token used for vector representation. Additionally, their exploration extended to cross-encoder models, known for success in the clinical domain [61]. In contrast to unsupervised semantic similarity, cross-encoders are trained by encoding both sentences simultaneously. The team employed two semantic search methods—BM25 ranking and transformer-based semantic search. In the latter, entity words and SNOMED CT were encoded with the SapBERT model, retrieving the closest candidate using cosine similarity. The predicted code for each entity derived from the most similar taxonomy entry. Their Semantic Search and Rerank (SS-R) methodology involved retrieving 64 candidates from SNOMED CT through semantic search or BM25, reranking them with the cross-encoder model, and selecting the candidate with the highest cross-encoder score. This chosen candidate's SNOMED CT code was then assigned to the respective entity. Meanwhile, in the Semantic Search and Conditional Post-Processing (SS-C) strategy, a nuanced exam-

ination of top similar items uncovered instances where the correct answer surfaced in the second position. This observation led to an enhancement of Accuracy K (K=2) by 5 points. By experimenting with similarity score thresholds on the development set, the team refined their approach, opting to select the second item if the first score fell below the established threshold.

As stated by [35], this SapBERT-XLMR model [62], known for its multilingual capabilities in representing biomedical concepts, was also employed by the Fusion team in an unsupervised manner. What is also noteworthy, the KFU team took a distinct route by training a model incorporating the Synonym Marginalization loss function [63]. Furthermore, they incorporated UniPELT adapters to enhance the efficiency of the model-fitting process.

Another instance worth highlighting involves a participant, Samy Ateia,[64] employing the GPT architecture, namely GPT 3.5-turbo and GPT 4. The overall performance of these systems did not exhibit notable strength across any of the three sub-tasks, especially with weak recall values. Nevertheless, when considering the submission in a broader context, it is likely that this system demanded less training compared to other approaches, primarily due to its fine-tuning through a few-shot learning approach. However, this advantage is somewhat offset by the high computational cost associated with GPT models. This insight points to the conclusion that BERT-based models outperform GPT-based models on the task of Entity Linking.

To summarize, the MedProcNER task at BioASQ 2023 reflects the dynamic evolution of Natural Language Processing (NLP) methodologies in biomedical contexts. Team Vicomtech's use of SapBERT and cross-encoder architectures demonstrates a robust approach to entity linking, enriched by multilingual clinical terminology. The KFU team's application of the Synonym Marginalization loss function and UniPELT adapters adds a new layer of novelty to the explored strategies. Additionally, Samy Ateia's introduction of the GPT architecture presents an uncommon perspective in EL, with utilizing GPT models This task provides crucial insights into the strengths and considerations of diverse NLP technologies, establishing benchmarks for medical concept recognition and entity linking.

Conclusions

In conclusion, the diverse range of papers discussed in this section underscores the rapid evolution and continuous innovation in the field of biomedical information extraction. There was a plethora of innovations and different approaches, beginning with the application of transfer learning by using pre-trained word embeddings as a foundation for training diverse

neural network models. This progression continued with the emergence of BERT-based models, models like BioBERT and ClinicalBERT, revolutionizing biomedical entity linking and achieving state-of-the-art results.

Researchers delved into the utilization of these BERT models fine-tuned specifically for the biomedical domain in entity linking tasks, leading to the introduction of models such as EhrBERT. This exploration emphasized the efficacy of fine-tuning domain-specific models for optimal performance. Additionally, SapBERT-XLMR pioneered the representation of biomedical concepts and their application in clinical entity linking. Beyond that, various approaches were proposed, ranging from the conventional two-step entity linking procedure involving candidate concept generation and subsequent reranking, to innovative models seeking combine these steps by employing a single transformer model. The candidate generation process came to be enhanced with semantic type prediction to challenge the problem of candidate concept overgeneration, yielding promising results. A distant supervision approach was introduced to tackle the challenge of data shortage, employing a rule-based approach for data augmentation. This strategy proved effective in mitigating issues related to human resource-intensive annotation processes.

Finally, significant strides were made in the domain of multilingual systems, utilizing machine translation to create a broader variety of biomedical corpora. Robust models like SapBERT-XLMR and BioM-BERT were trained to handle biomedical tasks in a variety of non-English languages, marking a noteworthy advancement in enhancing the inclusivity and applicability of biomedical information extraction methodologies globally. To provide a comprehensive comparison of the diverse methodologies discussed in the preceding papers, table 3.3 is presented below.

Despite all the significant advancements, challenges in biomedical NLP and entity linking still remain to be solved or improved upon, encompassing semantic ambiguity, data quality, lack of labeled data, interoperability, multilingual extraction, generalization to rare entities, adaptability to evolving terminology, and robustness to noisy data. Navigating the complexities of biomedical NLP, these challenges call for continued exploration and resolution, guiding the trajectory for the next phase of breakthroughs in this dynamic field.

Approach	Model/Architecture	Notable Results	Citation
Transfer Learning using word embeddings pretrained on EHRs and feeding them to NNs	CboW and CSG embeddings with FFN and RNN	RNN outperforms other models	Gligic et al. [18]
EL as a text classification task	Creation of EhrBERT from fine-tuning BioBERT of 1M EHRs	EhrBERT outperforms BioBERT and BERT, highlighting the importance of fine-tuning domain-specific models	Li et al. [24]
Two-staged EL approach, with IR based method and sentence-pair classification	fine-tuning BERT/BioBERT/ClinicalBERT	Results highlight the importance of fine-tuning domain-specific models	Ji et al. [57]
Semantic Type prediction for limiting candidate generation entities	Proposed MedType, DL system for semantic type prediction	Semantic type filtering enhances EL	Vashishth et al. [58]
Data augmentation with Distant Supervision	Distant supervision through a rule-based model. Single-step multi-label classification task using EhrRuBERT	The use of a single transformer was excelling in classifying common entities	Nesterov and Umerenkov [17]
SapBERT representations with cross-encoders	SapBERT-XLMR-large to embed entity words and SNOMED CT descriptions and XLM-RoBERTa trained on the UMLS.	Optimal results, leading the MedProcNER competition	Zotova et al. [60]

Table 3.3: Summary of Biomedical EL Approaches

Chapter 4

Idea and Approach

4.1 Data Exploration

Exploring data is a foundational step in uncovering valuable insights within datasets and corpora. The data explored in this section include a collection of EHRs (in particular discharge documents from cardiology wards) in the Greek language sourced from diverse hospitals across the country. This collection is accompanied by a structured dataset constructed by extracting medical term mentions from these records, each linked to its corresponding ICD-10 code. To enrich the dataset, the hierarchical structure of ICD-10 has been incorporated, including blocks and chapters. A deep understanding of this data is essential as it guides the methods applied in the subsequent sections and provides insights on the challenges encountered during the model-building process.

In this section, the exploration delves into the structured dataset, aiming to gain insights into the distribution and characteristics of medical mentions and their corresponding ICD-10 codes through the utilization of visual representations. Before delving into the actual findings it is important to understand the basic structure of ICD-10.

4.1.1 The structure of ICD-10

The International Classification of Diseases, 10th Edition (ICD-10), organizes diseases and health conditions into hierarchical structures for systematic coding. The classification is divided into chapters, each representing a broad category of diseases, and further subdivided into blocks. Chapters provide a high-level grouping, while blocks offer more specific categorization within each chapter. For instance, a chapter may focus on a particular system, such as the circulatory system, while blocks within that chapter could delineate specific conditions like hypertension. Understanding the hierarchy of ICD-10 codes, is crucial for accurate coding and enables a more nuanced analysis of health data. In the context of this study, the incorporation of the ICD-10 hierarchy enriches the data, providing additional layers of

information for a comprehensive exploration of the medical mentions found in the EHRs. Table 4.1 is provided to offer an overview of the chapters included in ICD-10.

Chapter	Code range	Description
1	A00-B99	Certain Infectious and Parasitic Diseases
2	C00-D49	Neoplasms
3	D50-D89	Diseases of the Blood and Blood-Forming Organs and Certain Disorders Involving the Immune Mechanism
4	E00-E89	Endocrine, Nutritional and Metabolic Diseases
5	F01-F99	Mental, Behavioral and Neurodevelopmental Disorders
6	G00-G99	Diseases of the Nervous System
7	H00-H59	Diseases of the Eye and Adnexa
8	H60-H95	Diseases of the Ear and Mastoid Process
9	I00-I99	Diseases of the Circulatory System
10	J00-J99	Diseases of the Respiratory System
11	K00-K95	Diseases of the Digestive System
12	L00-L99	Diseases of the Skin and Subcutaneous Tissue
13	M00-M99	Diseases of the Musculoskeletal System and Connective Tissue
14	N00-N99	Diseases of the Genitourinary System
15	O00-O9A	Pregnancy, Childbirth and the Puerperium
16	P00-P96	Certain Conditions Originating in the Perinatal Period
17	Q00-Q99	Congenital Malformations, Deformations, and Chromosomal Abnormalities
18	R00-R99	Symptoms, Signs, and Abnormal Clinical and Laboratory Findings, Not Elsewhere Classified
19	S00-T88	Injury, Poisoning, and Certain Other Consequences of External Causes
20	V00-Y99	External Causes of Morbidity
21	Z00-Z99	Factors Influencing Health Status and Contact with Health Services

Table 4.1: ICD-10 Chapter Descriptions

The chapters are subdivided into sections known as blocks, serving as subcategories within each chapter. These blocks, in turn, are detailed into base codes representing specific diseases. Within each base code, there are additional subcategories describing variations of the disease. To illustrate, an exemplification of the expansion of a block from the 9th chapter is presented in Figure 4.1

Chapter 9: Diseases of the Circulatory System (I00-I99)

- Block I20-I25: Ischemic heart diseases
  - I20: Angina pectoris
    - I20.0: Unstable angina
    - I20.1: Angina pectoris with documented spasm
    - I20.2: Refractory angina pectoris
    - ...
  - I21: Acute myocardial infarction
    - I21.0: ST elevation (STEMI) myocardial
      - I21.01: ST elevation (STEMI) myocardial infarction involving left main coronary artery
      - I21.02: ST elevation (STEMI) myocardial infarction involving left anterior descending coronary artery
    - I21.1: ST elevation (STEMI) myocardial infarction of inferior wall
- Block I26-I28: Pulmonary heart disease and diseases of pulmonary circulation
  - ...
- ... (other blocks within the chapter)

Figure 4.1: ICD-10 Structure Example

4.1.2 Data Analysis

The first step of the data analysis was the exploration of the distribution of different ICD-10 chapters and codes. The dataset extracted from the corpus contains in total 3,886 mentions corresponding to 199 different ICD-10 codes across 14 different chapters.

To gain insights into the distribution of medical codes in the dataset, Figure 4.2 presents a visual representation of the top 50 most frequently occurring ICD-10 codes. The bars are sorted in descending order, showcasing the most prevalent codes in the dataset. This analysis offers a snapshot of the predominant health conditions recorded within the electronic health records. In addition, Figure 4.3 was provided to illustrate the same distribution categorized by chapter.

Consulting the distribution of the ICD-10 chapters at Figure 4.4, it is evident that most codes come from chapter I. That is to be expected since the corpus comes from the cardiology wards of the hospitals, and with the use of Table 4.1 it is easy to see that the code ranges that focus on diseases of the circulatory systems are within the range of I00-I99. The other prominent chapter seems to be R, which is about symptoms, signs, and abnormal clinical findings not elsewhere classified. The predominant codes within this category, listed in order of

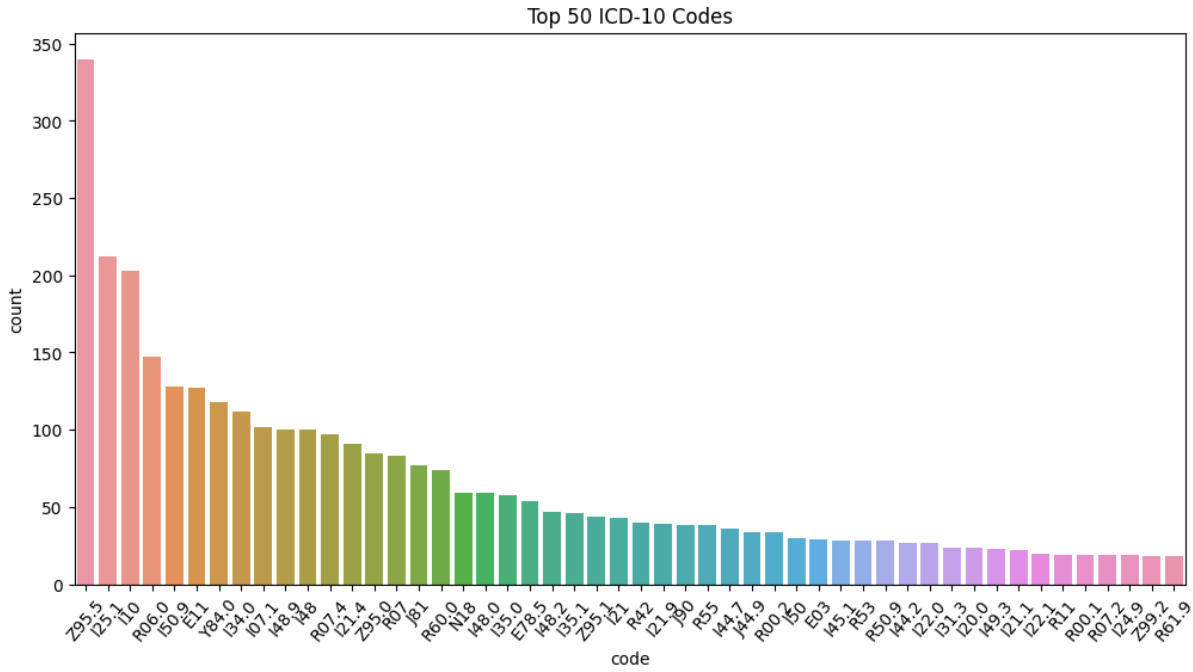

Figure 4.2: Distribution of the 50 most frequent ICD-10 codes

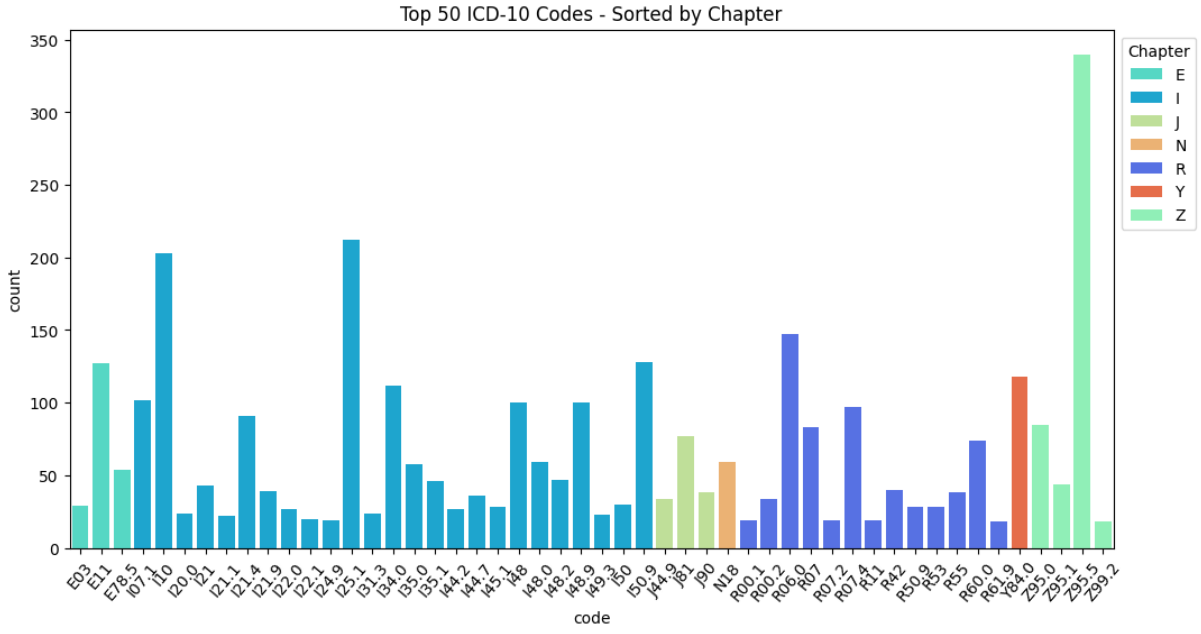

Figure 4.3: Distribution of the 50 most frequent ICD-10 codes divided by chapter

occurrence, include R06.0 denoting“signs and abnormal clinical and laboratory findings, not elsewhere classified,” R07.4 indicating unspecified chest pain, R60.0 representing “Localizes edema,” R42 associated with “Dizziness and Giddiness,” and R00.2 referring to heart palpitations. The prominence of this chapter underscores the diverse range of medical diagnoses observed, even within specialized cardiology wards.

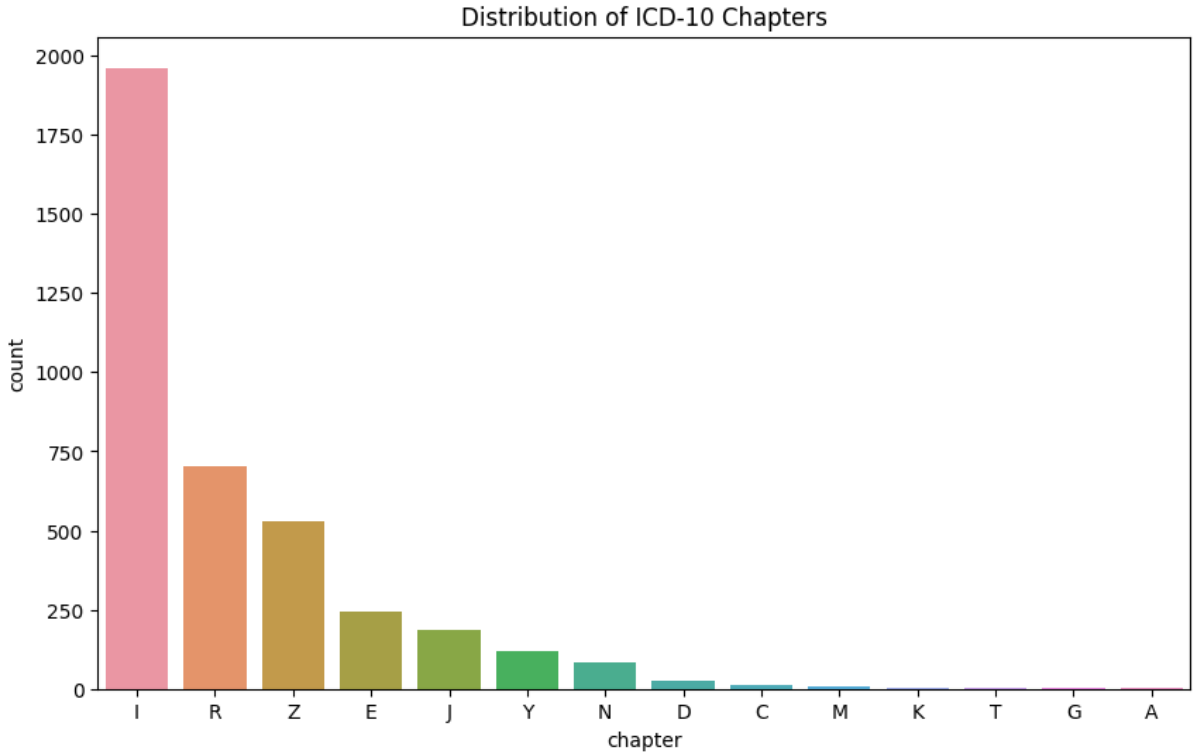

Figure 4.4: Distribution of ICD-10 Chapters

A noteworthy observation is chapter Z, ranking third in the number of occurrences. Chapter Z is about factors influencing health status and contact with health services. This prominence is primarily driven by the recurrent appearance of code Z95.5, which appears almost 350 times in the dataset, significantly surpassing the second-ranking code (I25.1) with just above 200 occurrences. The curious case of Z95.5 is in particular about the “presence of coronary angioplasty implant and graft”. Coronary angioplasty is a very common procedure and occurrence in the cardiology department, therefore the presence of such code is warranted.

The statistical summary of the occurrences of ICD-10 codes provides insights into the distribution pattern within the dataset. The dataset comprises 199 unique ICD-10 codes, with a mean occurrence of approximately 19.53. The variation in occurrences is notable, as indicated by the standard deviation of 40.58. The minimum occurrence is 1, reflecting codes that appear only once in the dataset. The interquartile range (IQR) provides further context, with the 25th percentile at 1, the median (50th percentile) at 4, and the 75th percentile at

18. This distribution suggests that a significant portion of codes occurs infrequently, with a median occurrence of 4. The statistics are summarized in Figure 4.2.

However, the maximum occurrence of 340 indicates the presence of codes that are highly prevalent in the dataset (Z95.5, as previously discussed), contributing to the observed variation. These statistical insights highlight the diversity in occurrences among the ICD-10 codes, emphasizing the importance of considering both the common and infrequent codes for robust model training and evaluation.

Statistic	Value
Count	199.00
Mean	19.52
Standard Deviation	40.57
Minimum	1.00
25th Percentile	1.00
Median (50th Percentile)	4.00
75th Percentile	18.00
Maximum	340.00

Table 4.2: Statistical Summary of ICD-10 Code Occurrences

4.2 Methodology

The proposed methodology draws inspiration from existing literature, embracing two distinct approaches commonly explored in the context of entity-linking tasks. The first approach aligns with the convention of employing a standalone classifier, tackling the entity linking task in a one-stage process. This strategy follows the work of Li et al. [24], who framed the problem as a text classification task utilizing BERT-based models. In addition to this, the proposed classifier capitalizes on the natural hierarchical structure of the ICD-10, earning the designation of a hierarchical classifier. This concept draws inspiration from the inherent structure of the ICD-10, which is constructed in a hierarchical manner and has a lot of information potentially encoded in it. To support that approach, in previous literature, a hierarchical structure was proposed by Bhargav et al. [53], where they propose a model that predicts not only the entity linked to a mention but also the hierarchical type of that entity in a logical hierarchy.

The second approach employs the two-stage approach of candidate concept generation and concept reranking. While this approach typically employs a bi-encoder model for candidate generation and a cross-encoder for reranking, the complexity of the cross-encoder led to a reevaluation of the strategy. The proposed approach combines the robustness of the bi-

encoder architecture for the candidate generation step with the proposed classifier, adhering to the principles of the two-stage entity linking (EL) approach. Simultaneously, it explores the capabilities of the hierarchical classifier during the reranking step.

For the candidate generation stage, a bi-encoder architecture is adapted, following the methodology outlined by Partalidou et al. [54]. This stage leverages the bi-encoder architecture to generate a set of candidate entities. Subsequently, the trained classifier is applied to rerank these candidates, providing a refined and optimized linkage of mentions to their corresponding concepts.

The standalone classifier approach was initially chosen due to the shortage of sufficient data and the limited availability of annotated ICD-10 codes within the dataset. The unique ICD-10 codes used for the annotations amounted to just 199 labels. This decision aimed to establish a baseline model and explore the capabilities of a BERT-based model in the task of entity linking for Greek medical texts, utilizing ICD-10 codes as the knowledge base. This architecture seeks to investigate how well a BERT-based model can adapt to the intricacies of Greek medical texts and assess the model's effectiveness in the entity linking task.

The choice of the bi-encoder architecture was motivated by the nature of the data. The mentions in the dataset contained numerous medical terms, often employing abbreviations or terminology specific to medical professionals. These terms might not align seamlessly with the descriptions present in the ICD-10 codes, making it challenging for traditional non-deep learning methods to work effectively. The training process of the bi-encoder embeddings enables the model to learn the nuanced similarities and dissimilarities of mention-context pairs. This is achieved through the utilization of a contrastive loss function, which guides the training of embeddings to better capture semantic relationships. The bi-encoder model's effectiveness is further assessed by comparing it to traditional string similarity metrics. This approach acknowledges the unique challenges posed by the dataset's linguistic intricacies and demonstrates the flexibility of the bi-encoder architecture in capturing semantic associations that might be elusive for conventional methods. While the bi-encoder architecture is expected to aid in generating a focused set of candidate entities for the entity-linking classifier, the impact on accuracy is currently exploratory, and results will be analyzed to determine its efficacy.

The two architectures are evaluated both jointly and autonomously. The hierarchical classifier, functioning as a standalone classifier, is systematically compared to alternative classifier configurations for the 199 labels, aiming to identify optimal settings for Greek EHR entity linking. The hierarchical classifier was also evaluated jointly with the use of the bi-encoder

performing the candidate generation step, and finally, the bi-encoder underwent an independent evaluation of its ability to retrieve the correct entity within the top-k retrieved candidate concepts.

4.2.1 Hierarchical Classifier

Overview

The proposed model incorporates a hierarchy-enforced classifier that utilizes the discussed blocks of the ICD-10 structure as part of the loss function. This classifier integrates contextual embeddings from BERT with a specialized mention pooling function, considering custom special tokens to capture relevant mention information.

To capture contextual information, contextual embeddings from BERT are employed. BERT's pre-trained language representations provide a rich understanding of words in a given context for natural language text. This ability to discern the meaning of various terms, could easily extend to the nuanced nature of medical texts. The proposed model incorporates a special token pooling function, which plays a pivotal role in aggregating information from contextual embeddings.

A unique feature of the proposed model is the incorporation of a Hierarchical classifier. This classifier utilizes specific blocks of the ICD-10 hierarchy as part of the loss function, and one separate classification layer for each hierarchy level, allowing the model to consider not only direct semantic relationships but also hierarchical associations among medical entities. The hierarchical loss function was implemented by combining losses from block labels (parent classes) and the actual ICD-10 codes (child classes) as annotated by medical professionals. This dual approach aims to reinforce both the broad categorization and the specific code assignments, aligning the model with the hierarchical structure of the ICD-10 taxonomy. The proposed architecture is summarized in figure 4.5

Input representations

The input data underwent meticulous processing to extract mentions from the text, accompanied by left and right context. Context extraction involved utilizing a predefined number of tokens on each side of the mention to provide comprehensive contextual information. Each input representation was formatted with special tokens, explicitly designed to offer context and outline mention boundaries. To guide the tokenizer in correctly processing these representations, additional special tokens, $[Ms]$ (indicating the start of a mention) and $[Me]$ (indicating the end of a mention), were introduced. Considering the BERT tokenizer's de-

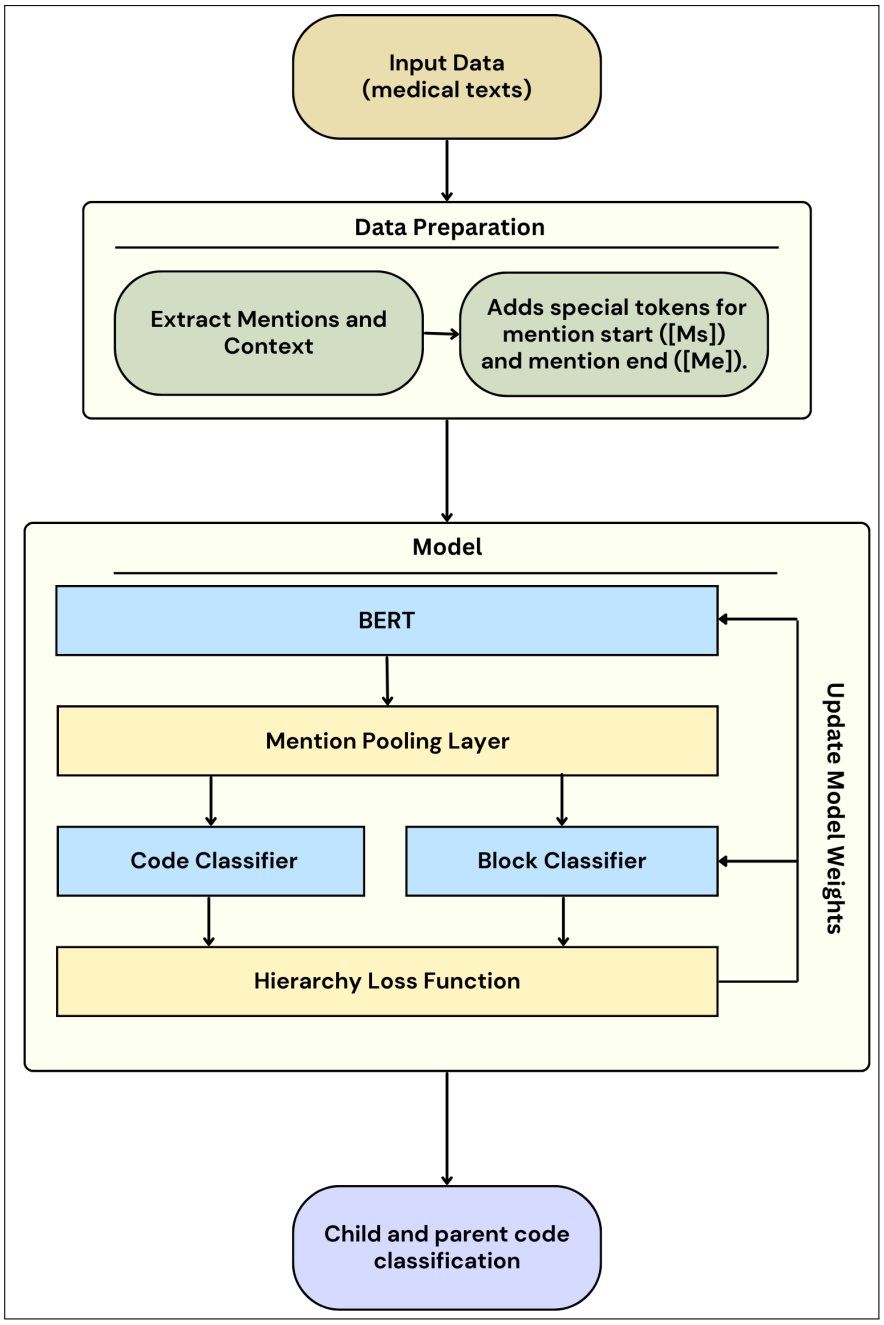

Figure 4.5: Hierarchical Classifier Architecture

fault special tokens $[CLS]$, which is a special token used by BERT at the beginning of each input sequence, carrying information about the entire sequence and providing a summary representation, and the $[SEP]$ special token, used to mark the separation between different segments or sentences, the final format of the input representations looks as follows:

$$[CLS]\ context\_left\ [Ms]\ mention\ [Me]\ context\_right\ [SEP]$$

These special tokens were employed upon the hypothesis that they would assist in guiding the tokenizer to handle mentions within the text appropriately. To ensure seamless integration

with the tokenization scheme adopted for the entity linking task, the token embeddings of the model were resized. This step is crucial to align the model's internal token embeddings with the token indices generated by the tokenizer during input processing.

Pooling functions

For the step of the input representation pooling, two strategies were tested:

- Usage of $[CLS]$, the default representation as a summarization of the entire sequence and BERT's default *pooler_output*:

$$pooler\_output = Tanh(Linear(CLS\_hidden\_state))$$

  Where in this formula, CLS_hidden_state is the last layer hidden-state of the first token (classification token) of the sequence, $Linear$ represents the linear layer, and the weights of this layer are trained during pre-training for the next sentence prediction (classification) objective. $Tanh$ is the hyperbolic tangent activation function.

- Average special, averaging only the special token representations of $[Ms]$ and $[Me]$ from BERT's last hidden layer.

$$\text{Average special} = \frac{1}{N}\sum_{i=1}^{n} embedding[i]$$

  This formula represents the average pooled representation, where N is the total number of tokens in the sequence, n is the number of $[Ms]$ and $[Me]$ tokens (2), and $embeddings$ denotes the extracted embedding of the i-th token, which in this case represents the embeddings corresponding to the $[Ms]$ and $[Me]$ special tokens.

Code and Block classifiers

As mentioned, the model is structured to perform classification for the annotated codes (child classes) and their corresponding blocks (parent classes). The classification layers are implemented as linear layers, transforming the hidden representations obtained from the BERT model into logits for each class, given the representations acquired by the pooling functions. The *parent classifier* linear layer is designed for hierarchical block classification, with the number of output units equal to the unique number of block labels. Similarly, the *child classifier* linear layer aims at predicting specific ICD-10 codes, and its output units match the number of unique code labels in the dataset.

Loss Function and Optimization

Bhargav et al. [53] introduced a novel approach by incorporating a logical framework that leverages the probabilities of nodes along hierarchical paths, helping the model to predict entity types with logical consistency. To implement hierarchical information in this study, a simplified approach that considers hierarchical path probabilities is adopted. The proposed loss function is a combination of losses from both the block (parent) and code (child) classifiers. The principle with which the combined loss was calculated follows this formula:

$$loss = (loss\_parent * parent\_weight) + (loss\_child * child\_weight)$$

Where *loss_parent* and *loss_child* represent the losses acquired from each classifier using a *Cross Entropy Loss* criterion and their corresponding weights are employed to control the training process and prevent overfitting over a certain hierarchy level.

The optimization step involves backpropagating the gradients through the entire model and updating the parameters using the specified optimization algorithms. In this case, the same loss is used to update the weights of all three components (BERT and child and parent classifiers). To provide flexibility in training dynamics, the model includes training control flags *(trainBERT*, *trainParent*, *trainChild)* that allow for selective freezing of specific components during training. When a particular flag is set to False, the corresponding parameters of the associated component are frozen. This allowed for the model to be tested for several configurations, and dynamically control overfitting combined with the loss weights.

Model Outputs

As a result of this methodology, the proposed model is suited for block code predictions, annotated code predictions and is capable of producing predicted probabilities for both hierarchy levels, which provide the option for the same model to be used for the concept reranking stage. In the current implementation, the block and code predictions are independent of each other, without including the case of there being a discrepancy between them. This aids in evaluating the proposed loss function without introducing additional parameters to the training process. It would be beneficial however to examine how the model performs when considering a strict hierarchical structure by resolving the potential discrepancies.

4.2.2 Bi-Encoder

Overview

The bi-encoder architecture that complements the implementation consists of two separate BERT-based transformer instances, one meant for encoding the mentions and one meant for encoding the codes and their respective descriptions. In conjunction with BERT embeddings, the model incorporates pooling functions for both mention and entity embeddings. During training, a contrastive loss function is employed to capture the similarities and dissimilarities within each mention-concept pair. An overview of the architecture is provided in Figure 4.6

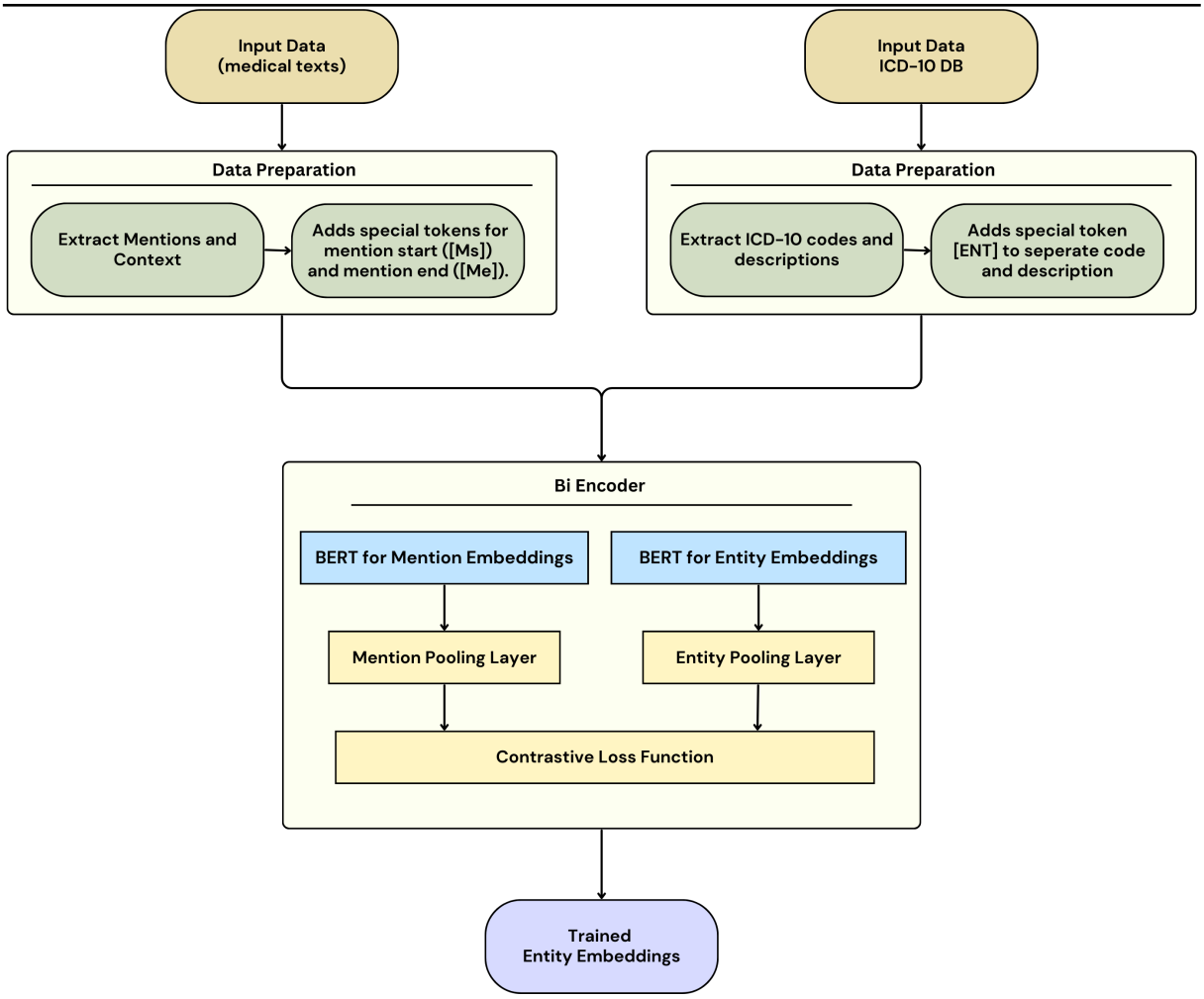

Figure 4.6: Mention-Entity Bi-Encoder Architecture

Input Representations

The input data for the bi-encoder comprises mentions extracted from medical texts, accompanied by their corresponding ICD-10 codes and respective descriptions. For the input repre-

sentations of the bi-enocder, the same methodology as the hierarchical model was employed for the mention inputs, utilizing context extraction and the presence of introduced special tokens $[Ms]$ and $[Me]$. In addition, a new special token, $[ENT]$ was introduced, to act as a separation barrier between the ICD-10 codes and their descriptions. So in summary, the input representations of the mentions and entities for the bi-encoder are as follows:

- $[CLS]$ $context\_left$ $[Ms]$ $mention$ $[Me]$ $context\_right$ $[SEP]$
- $[CLS]$ $ICD10\_code$ $[ENT]$ $Code\_description$ $[SEP]$

Pooling functions

For the step of the input representation pooling, two strategies were tested:

1. Usage of [CLS]: Situated at the beginning of the sequence, it is employed as the default representation summarizing the entire input sequence. This option utilizes the [CLS] token representation for both mention and entity embeddings.

$$\text{CLS} = \textit{Representation of the [CLS] token in the last hidden state}$$

2. Extracting special token representations:
   - Average Special: Averaging only the special token representations of $[Ms]$ and $[Me]$ from BERT's last hidden layer. This method is specifically used for mention embeddings.

$$\text{Average Special} = \frac{1}{n}\sum_{i=1}^{n}\text{embedding}[i]$$

   - For entity embeddings, as only one token is extracted, averaging is not applied.

$$\text{ENT} = \textit{Representation of the [ENT] token in the last hidden state}$$

Loss function

The choice of using contrastive loss in this context is motivated by the desire to effectively capture the semantic relationships between mentions and entities. Contrastive learning is a paradigm in machine learning that aims to train models to distinguish between positive pairs (instances that should be similar) and negative pairs (instances that should be dissimilar). This is achieved by pulling similar instances closer in the embedding space while pushing

dissimilar instances apart. In the context of entity linking, positive pairs correspond to correct mention-entity pairs, where the mention should be linked to the specific entity. Negative pairs, on the other hand, consist of mentions and entities that should not be linked. The model is trained to minimize the distance between embeddings of positive pairs and maximize the distance between embeddings of negative pairs. Contrastive loss contributes to the learning process by enforcing the model to discern the nuances in the relationships between mentions and entities. By explicitly considering both positive and negative pairs, the model is encouraged to create meaningful embeddings that capture the semantic context and relationships within the medical texts.

The similarity score of the entity candidate e_i given a mention m is computed by the dot-product:

$$s(m, ei) = embedding\_m * embedding\_ye_i$$

For each training pair (m_i, e_i) in a batch of B pairs, the contrastive loss is computed as:

$$Loss(m_i, e_i) = -s(m_i, e_i) + \log\left(\sum_{j=1}^{B} \exp(s(m_i, e_j))\right)$$

Candidate Generation Step

The bi-encoder is used for the candidate generation step, by providing embeddings for the ICD-10 codes and their descriptions and the specified mention. This process involves pre-computing embeddings for the ICD-10 codes and their descriptions by using a forward pass of the model, where the input is in the form of

$$[CLS]\ ICD10\_code\ [ENT]\ Code\_description\ [SEP]$$

Subsequently, for each mention under consideration, the mention also undergoes a pass through the bi-encoder to obtain its embedding. The similarity metric between the mention embedding and each ICD-10 code embedding is then computed using a specified similarity metric. Ultimately, the top-k codes are retrieved, forming the candidate concept set for further entity-linking tasks. This process is described in Figure 4.7.

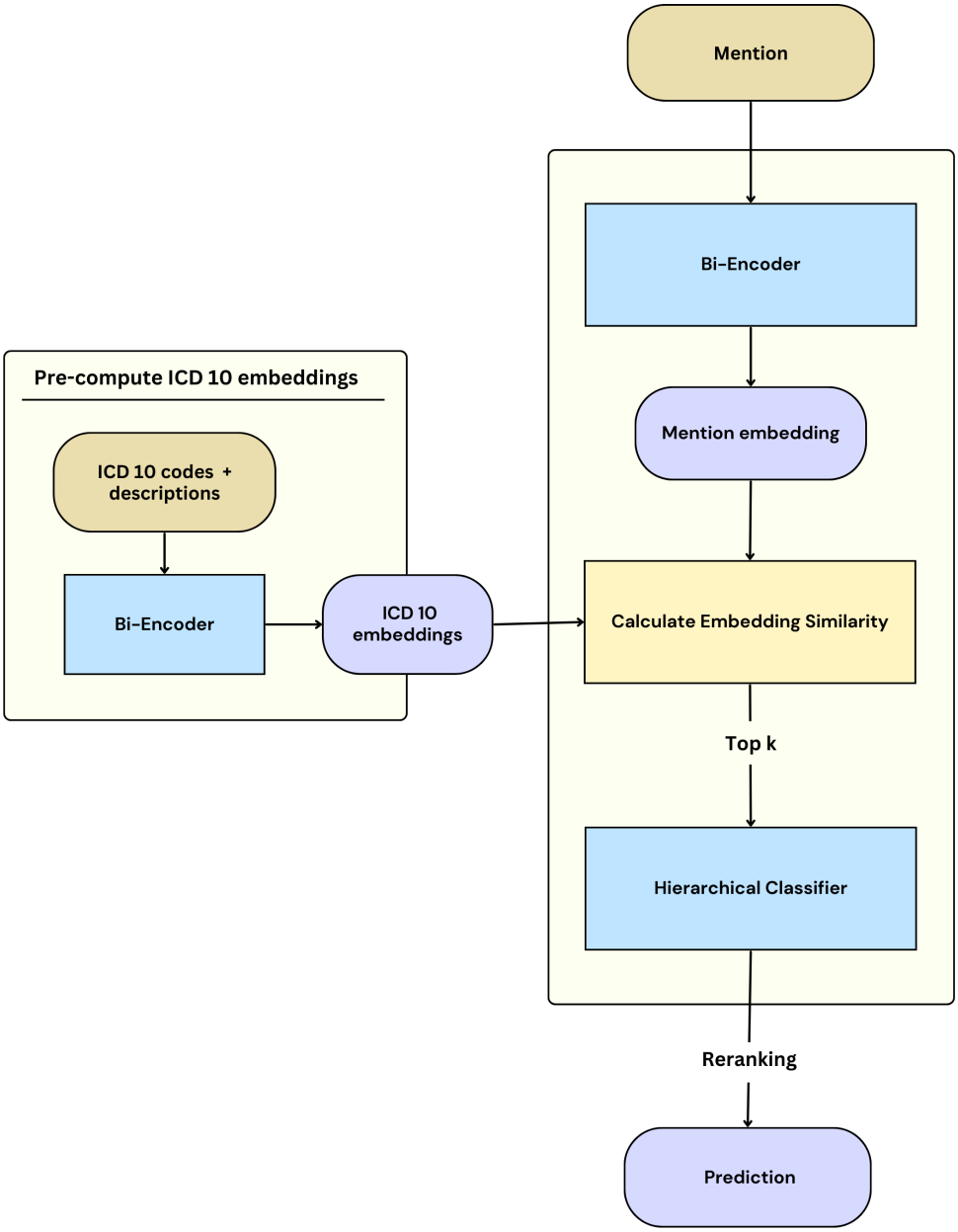

Figure 4.7: Two-staged EL combining the bi-encoder and the hierarchical cls

Model Outputs

The training objective of the bi-encoder model centers on the generation of embeddings adept at encapsulating both the similarity and dissimilarity aspects within mention-entity pairs. The key model outputs include:

- **Encoded Mention**: This output represents a condensed and contextually rich embedding capturing the semantic content of extracted mentions from medical texts.
- **Encoded Entities**: The model produces embeddings for each candidate entity in the

knowledge base, providing a contextual representation of their descriptions.

- **Similarity Calculation**: Utilizing the encoded mention, the model calculates similarity scores with all entity embeddings. This step is crucial for identifying entities with contextual resemblance to the given mention.
- **Candidate Set Generation**: The final step involves creating a candidate set of entities based on the calculated similarity scores. This set serves as the initial pool for the subsequent entity classification stage.

Chapter 5

Implementation and Experimentation

In this chapter, the focus shifts to the practical aspects of the research, providing an overview of the implementation details and the systematic experimentation carried out to evaluate the proposed methodologies. The chapter commences by outlining the dataset creation process, starting from the unprocessed discharge documents and navigating through the annotation phase to achieve a final structured format. Subsequently, it proceeded to describe the implementation process, encompassing data pre-processing procedures and a comprehensive examination of both the Hierarchical classifier and bi-encoder implementations.

5.1 Creation of the Dataset

The dataset creation process involved a collaborative effort between our team and a team of medical professionals. To facilitate the annotation process, the doccano annotation tool was installed on a dedicated server, providing the doctors with a user-friendly platform for performing annotations.

5.1.1 Document De-identification

The first step of the process was for the discharge documents to be de-identified. To better understand the de-identification process, it would be helpful to provide the general structure of the discharge documents. All the discharge documents used for this study followed a similar format, which is portrayed in Figure 5.1. They were separated into three broad sections, the first and last of which contained personal information about the patient (name, age, social security number, admission, and discharge dates) and the names of the medical personnel in charge. Those two sections were removed completely to protect the personal information of both the patients and the doctors involved.

To further ensure the correct de-identification of the documents, a thorough review of the central section containing details about patient examination, admission progression, and

Section	Contents
1	Hospital Information
2	Patient information
3	Reason for Admission - Objective Examination - Patient History
4	Course of Illness for Admission Duration
5	Lab Exams
6	Discharge Diagnosis
7	Discharge Instructions and Observations
8	Medical Personnel Names

Figure 5.1: Structure of a typical discharge document

discharge instructions was conducted. This scrutiny aimed to eliminate any residual instances of patient names, doctor names, or their contact information, ensuring a meticulous cleansing of sensitive data. That examination was deemed necessary as it was found that in some occasions, doctor's names were referred to in section 4 of Figure 5.1, and instances of both doctor names and phone numbers were detected in Section 7, where patients were referred for additional examinations. Furthermore, any more references to dates and hospital names were also removed from the text.

5.1.2 The Annotation process

The annotation process followed a structured workflow, with the medical professionals systematically reviewing and annotating the medical texts using predefined guidelines. These guidelines ensured consistency and accuracy in the annotations across the dataset. Additionally, regular meetings and discussions were held between our team and the medical professionals to address any ambiguities or discrepancies in the annotations and to refine the annotation process further.

The initial set of documents slated for annotation comprised 1,000 documents, which were evenly divided into two sets. These sets were then assigned to two separate teams of medical professionals, each composed of two individuals. The annotation process involved independent work, with each team member annotating their assigned set without access to the annotations made by their counterpart. In the final annotation stage, the agreement between annotators was taken into consideration. Annotations that both annotators agreed upon were considered correct, while annotations with discrepancies were individually examined and addressed. Figure 5.2 provides a visual representation of the annotation process within the

doccano environment. The figure illustrates annotations made by both annotators, with two highlighted spans indicating annotator agreement by assigning the same label to the respective spans. Additionally, the figure highlights a mention that was overlooked by one annotator but correctly identified by the other.

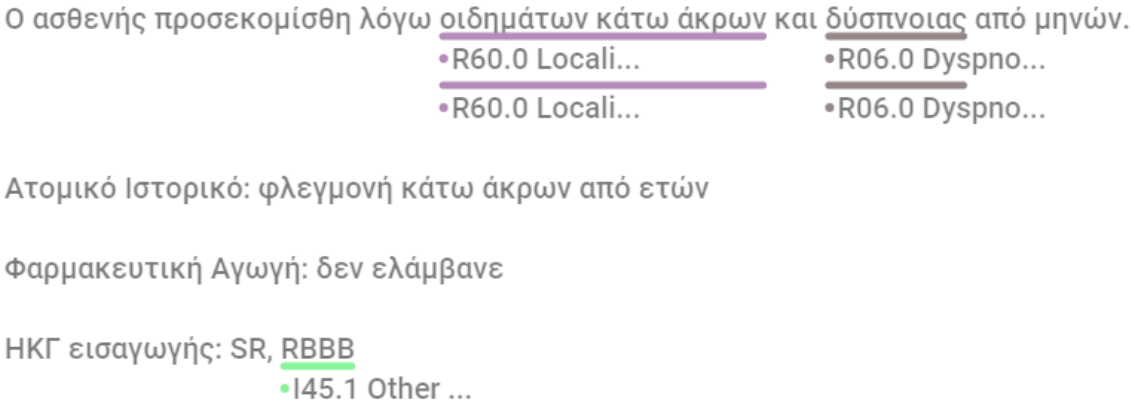

Figure 5.2: Anootation prosses on the doccano environment

At the initiation of this study, only 382 out of the 1,000 discharge documents had been annotated, presenting limitations to the scope of the research. However, the 3,886 mentions identified within these 382 annotated documents provided a sufficient basis for the initial exploration of the capabilities of the proposed models. It is anticipated that the annotation process will be finalized in subsequent phases of the research.

5.1.3 Shaping the Final Dataset Form

After completing the annotation of 382 documents, each mention was extracted and formatted into a CSV file. To distinguish between documents, a unique identifier, referred to as the 'patient ID,' was assigned. All mentions sharing the same patient ID originated from the same document. The 'ICD 10 code' column represents the annotated code for each mention, while the subsequent three columns display the starting and ending positions of the mention, along with the actual text. Figure 5.3 provides an illustrative example.

Patient ID	ICD 10 code	Start	End	Mention
175	I79.0	415	439	Διάταση ανισούσας αορτής

Figure 5.3: Tabular dataset form

For the purpose of training context-aware models, the dataset was complemented by the full text of each document. Each text was uniquely identified using the 'patient ID' number. By matching mentions with the patient ID identifier and utilizing the start and end positions

of each mention, the relevant context was extracted from the texts. This process ensured that the context surrounding each mention could be effectively incorporated into the training data.

5.2 Implementation

5.2.1 Data Preprocessing

The data preprocessing step involves the extraction of the mentions and their contextual information, followed by using the Wordpiece model through the BERT Tokenizer. These steps lay the foundation for subsequent stages of the model, ensuring that the information-rich content within the medical texts is appropriately represented for further analysis.

The mentions were extracted from the discharge letters using the mention start and end indexes. Surrounding contexts, to the left and right of the mention, were extracted based on a predetermined number of tokens, where tokens correspond to complete words. During model training, a context window of five words on each side was chosen for effective contextual information capture. The choice of a 5-word context window was determined through data analysis, considering that beyond this window, the text might contain less relevant information for the entity linking task.

The tokenizer was initialized using a BERT tokenizer that was initialized using the pre-trained model bert-base-multilingual-uncased, and the tokens *[Ms]*, *[Me]* and *[ENT]* were introduced as additional special tokens. The specified model refers to a version of BERT that supports multiple languages (multilingual) and is case-insensitive (uncased), meaning it treats uppercase and lowercase letters the same way during tokenization. This tokenizer is pre-trained on large datasets and can be used to convert text into numerical tokens that can be processed by the BERT model. The choice of the BERT multilingual model was deliberate, as it provides support for the Greek language, having been also trained on Greek Wikipedia. Additionally, the model demonstrates proficiency in handling English words that may appear in the text. The tokenizer utilizes the WordPiece model, a subword tokenization technique that breaks down words into smaller units, enhancing its ability to represent a wide range of languages and handle variations in morphology, even for words that don't appear in its vocabulary, as shown in Table 5.1.

When tokenizing the text to include context, a maximum length of 128 tokens was implemented to accommodate longer texts. In experiments where only the mention was used as the model input (no-context experiments), no special tokens ([Ms] and [Me]) were included, and a maximum length of 32 tokens proved sufficient. Longer sequences were truncated to meet the specified maximum length. Examples of mentions and their corresponding left and

Mention	Wordpiece representation
Δύσπνοια - ορθόπνοια	[‘δ’, ‘υ’, ‘σ’, ‘π’, ‘νο’, ‘ια’, ‘-’, ‘ο’, ‘ρ’, ‘θ’, ‘ο’, ‘π’, ‘νο’, ‘ια’]
NSTEMI	[‘NS’, ‘TE’, ‘MI’]
υπερτασικής κρίσης	[‘υ’, ‘πε’, ‘ρτ’, ‘α’, ‘σι’, ‘κης’, ‘κ’, ‘ρισ η’, ‘ς’]

Table 5.1: Wordpiece Examples

right contexts, along with their representations after they’ve been tokenized, are provided in Table 5.2.

Mention	Left Context	Right Context	Encoded Text
Μεγαλοκαρδια	Συμφορηση πυλων. Επιταση διαμεσου δικτυου.	Υπερηχος καρδιας ΑΡ κοιλια με	[CLS] συμφορηση πυλων. επιταση διαμεσου δικτυου. [Ms] μεγαλοκαρδια [Me] υπερηχος καρδιας αρ κοιλια με [SEP]
ΑΫ	Α/α θώρακα: ΚΦ Ατομικό Αναμνηστικό:	ΠΟΡΕΙΑ ΝΟΣΟΫ Εισαγωγή ως NSTEMI.	[CLS] α / α θωρακα : κφ ατομικο αναμνηστικο : [Ms] αυ [Me] πορεια νοσου εισαγωγη ως nstemi. [SEP]
στένωση μιτροειδούς	αριστερά. Ατομικό Αναμνηστικό: Ρευματική βαλβιδοπάθεια:	μεικτή βλάβη αορτικής. Κολπική	[CLS]ατομικο αναμνηστικο : ρευματικη βαλβιδοπαθεια : [Ms] στενωση μιτροειδους [Me], μεικτη βλαβη αορτικης. [SEP]

Table 5.2: Mention and context examples

For the entity texts used in the bi-encoder, a consistent tokenization approach was adopted. The same BERT tokenizer, initialized with ‘bert-base-multilingual-uncased,’ was employed to maintain uniformity in tokenization. The full Greek description of each ICD-10 code in the dataset was utilized, and it was observed that a maximum length of 128 proved sufficient for comprehensive representation.

5.2.2 Hierarchical Classifier

The hierarchical classifier is implemented by initializing a BERT-based model using the ‘bert-base-multilingual-uncased’ pre-trained model. As explained in the previous section, the model was selected due to its multilingual capabilities, in order to properly handle texts

in the greek language as well as the English words that may be incorporated inside. The token embeddings of the model are resized to match the vocabulary size of the pre-discussed tokenizer, enlarging the vocabulary from 105,879 words to 105,881. This resizing step is crucial for aligning the model's internal token embeddings with the token indices generated during input processing. Standard back-propagation is used to update all the parameters and an Adam optimizer with a learning rate of 1e-5 to control the update process.

Two linear classifiers, one for parent (block) classifications and one for child (code) classifications, are initialized based on the BERT model's hidden size and the unique number of block and code labels in the dataset. The Cross Entropy Loss criterion is chosen for computing classification losses. Separate Adam optimizers are set up for updating the parameters of the parent and child classifiers during training. The activation function used in the linear classifiers is softmax, which converts the raw output scores into probabilities. In the parent classifier, these probabilities represent the likelihood of a medical entity belonging to a specific block, while in the child classifier, they indicate the probability of a medical entity being associated with a particular ICD-10 code.

The losses of each linear classifier are obtained through the cross-entropy criterion to be accumulated into the final loss as described by $loss = (loss\_parent * parent\_weight) + (loss\_child * child\_weight)$. This final loss is then utilized in the backpropagation step to update the weights of the BERT model and the parent and child classifiers, employing the Adam optimization algorithm. Before settling on the final hyperparameters, an exploration of various values for learning rates (ranging from 5e-5 to 5e-6), optimizer decaying rates, hierarchical loss function ratios (with parent weight varying from 0.5 to 1), mention context window, and embedding size was conducted. Results for different learning rates are shown in Table 5.3.

Learning Rate		**Accuracy**	**Precision**	**Recall**	**F1**
5e-5	(parent)	90.48%	54.34%	53.57%	53.58%
	(child)	78.56%	48.97%	51.56%	48.79%
3e-5	(parent)	90.87%	51.29%	53.29%	51/70%
	(child)	76.89%	47.08%	49.64%	46.76%
1e-5	(parent)	**95.11%**	**72.22%**	**69.83%**	**70.17%**
	(child)	**81.98%**	**60.65%**	**62.92%**	**60.34%**
5e-6%	(parent)	90.86%	70.00%	66.85%	67.82%
	(child)	80.05%	54.67%	56.99%	54.31%

Table 5.3: Learning Rate Results for the context-aware Hierarchical classifier

The optimal context window size, determined to be relatively small due to high variations in text where context becomes less relevant in longer sections, and an embedding size of 128,

suitable for the full sequence of mention and context, were selected. For mentions without context, a size of 32 proved sufficient. Refer to Tables 5.4 and 5.5 for a summary of the hyperparameters used in the classifier architectures.

Hyper Parameter	**Value**
Batch size	32
BERT Learning rate	1e-5
Linear layer lr	1e-3
Dropout	0.1
BERT optimizer	Adam
Linear layer optimizer	Adam
Linear layer criterion	Cross Entropy Loss
Activation function	Softmax

Table 5.4: Common hyperparameters shared by all classifier models.

As a result of the hierarchical classifier methodology, the proposed model is tailored for block code predictions, annotated code predictions, and is capable of producing predicted probabilities for both hierarchy levels. This flexibility allows the same model to be used for the concept reranking stage, providing a comprehensive solution for medical entity prediction and categorization. The entire architecture is encapsulated in an instance of the HierarchyClassifier class, providing a cohesive structure for hierarchical predictions with fine-tuned control over the model's components and optimization.

For a thorough evaluation of the proposed model, various configurations were employed during the training phase and systematically compared:

1. **Baseline Model**: this configuration of the hierarchical model was set up to act as a baseline classifier, utilizing only a simple loss function by setting the weight of the parent loss to 0, and freezing the block linear classifer. This resulted in a model that performs classification only for the annotated codes, without considering the hierarchy of the ICD-10. In this baseline model, no context was taken into consideration.

2. **Context Aware Baseline classifier**: This configuration is similar to the previously described baseline model, with the difference that a context window of 5 tokens on each side of the mention was included in the model's input. This adjustment aimed to enhance the model's classification performance by considering the surrounding textual context during the prediction process.

3. **Hierarchical Classifier without context**: This initial variation of the Hierarchical classifier is designed without a context window, utilizing only the mentions as inputs. While

acknowledging the hierarchical structure of the ICD-10, this model incorporates the loss from the block classifier into the loss function. The loss function is formulated as *parent loss * 0.9 + child loss * 1*. This specific ratio was determined to effectively mitigate overfitting for the parent classes, allowing the model to gradually learn the child classes.

4. **Context Aware Hierarchical Classifier**: Similar to the preceding configuration of the hierarchical classifier and following the same pattern as with the baseline classifiers, this variant includes a context window of 5 tokens on each side of the mentions in the input data.

A summary of each model's configuration is provided in Table 5.5.

Parameters	**Base Cls**	**Context Aware Base Cls**	**Context free Hierarchical**	**Context Aware Hierarchical**
BERT word embeddings	(105879, 768)	(105881, 768)	(105879, 768)	(105881, 768)
additional special tokens	-	[Ms], [Me]	-	[Ms], [Me]
pooling function	CLS	Special Avg	CLS	Special Avg
Backpropagation loss	child loss	child loss + 0.9*parent loss	child loss	child loss + 0.9*parent loss
mention embeddings	32	128	32	128
num window words	-	5	-	5
num epochs	5	5	6	6
Linear Layer Parameters				
Parent Classifier Units	-	-	48	48
Child Classifier Units	199	199	199	199

Table 5.5: Model architecture and training parameters for different classifier configurations

5.2.3 ICD-10 Bi-Encoder

For the bi-Encoder architecture, leveraging the same multilingual model, two separate BERT-based transformer instances were initialized. One dedicated to mention encoding and the

other to entity encoding. To ensure seamless integration with the chosen tokenizer and maintain consistency, the token embeddings of both transformers were resized based on the vocabulary size of the tokenizer. To facilitate fine-tuning during training, separate optimizers were defined for each one of the transformers, both using a different instance of an Adam optimizer. The implementation further employed a contrastive loss function during training, designed to capture both similarities and dissimilarities within each mention-concept pair. This approach aimed to enhance the generation of effective embeddings for subsequent retrieval tasks. The weights of both transformers are updated during backpropagation using the contrastive loss and an Adam optimizer.

5.3 Experimentation Setup

5.3.1 Dataset

The proposed model was evaluated on a dataset consisting of medical texts with annotated ICD-10 codes. The dataset includes a range of diverse hospital discharge documents, specializing in the cardiology department. The annotations provide information about the relevant hierarchical blocks and specific ICD-10 codes associated with medical entities mentioned in the text. Table 5.6 provides a summary of the dataset's sets distribution. This first part of the dataset was to train the classifier models and train the mention embeddings of the bi-encoder model.In the context-aware configurations, an input window of 5 words on each side of the mention was incorporated into the input data. Additional distinctions among various model configurations concerning the input data are outlined in Table 5.5.

The second part of the training data involves the descriptions of the ICD-10 codes found in the dataset, specifically addressing the training of the entity embeddings within the bi-encoder. The training set, as illustrated in Table 5.6, served as the basis for training the bi-encoder. To complement the dataset, detailed descriptions obtained from the official website of the Greek Ministry of Health[1], corresponding to the 199 ICD-10 codes, were incorporated into this set. These descriptions, being official and authoritative, provide comprehensive insights into the medical conditions represented by each ICD-10 code.

An example of the input data for the bi-encoder model is presented below, illustrating the relationship between mentions and their corresponding entities.

[1]https://www.moh.gov.gr/articles/health/domes-kai-draseis-gia-thn-ygeia/kwdikopoihseis/kleista-enopoihmena-noshlia/713-kwdikopoihseis

Mention	Entity
..ανέδειξε επιδείνωση της γνωστής [Ms] καρδιακής ανεπάρκειας [Me] , με διαστολική δυσλειτουργία...	I50.9 [ENT] Καρδιακή ανεπάρκεια, διάφορες περιπτώσεις

Set	N^o of mentions	N^o of unique labels
Entire Dataset	3,886	199 codes 48 blocks
Train set	2,720	183 codes 46 blocks
Val set	389	88 codes 24 blocks
Test set	777	124 codes 35 blocks

Table 5.6: Contents of train, test and validation sets

5.3.2 Models

The classifiers are trained with a batch size of 32, using the Adam optimizer with a learning rate of 1e-3. The base models are trained for 5 epochs, while the hierarchical models undergo training for 6 epochs. The chosen number of epochs ensures convergence and optimal performance. The hyperparameters are outlined in more detail in Tables 5.5 and 5.4.

The bi-encoder model is trained with a batch size of 8 due to memory limitations, utilizing the Adam optimizer with a learning rate of 1e-5. Training for 5 epochs ensures the model adequately captures mention and entity embeddings. Additional training details are outlined in Table 5.7.

5.3.3 Evaluation

The models underwent a comprehensive evaluation using established metrics, covering the following settings:

- Evaluation of the hierarchical classifier as a standalone classifier with various configurations to identify the optimal model.
- Evaluation of the bi-encoder in collaboration with the hierarchical classifier, implementing a two-staged EL process involving candidate generation and reranking steps.
- Evaluation of the bi-encoder independently, without the influence of the classifier.

- Evaluation of the models with document-level labels.

Metrics considered for evaluation included accuracy, precision, recall, and F1 scores, calculated using macro-averaging to provide a comprehensive assessment. For the hierarchical classifier, assessments were made based on its ability to predict both parent and child classes. The bi-encoder was evaluated in conjunction with the hierarchical classifier, analyzing how the candidate generation step affected the classifier's performance using the same evaluation metrics.

Candidates were retrieved using different similarity measures: dot product, cosine similarity, and Euclidean distance. Top-k retrieval evaluations were conducted across various values of k. These results offer insights into the models' performance across different configurations, providing a comprehensive understanding of their strengths and limitations.

To apply similarity metrics, embeddings of the 199 code descriptions from the entire set were initially extracted through a forward pass. Subsequently, for each mention in the test set, its embedding was extracted, and the similarity between the mention and each of the 199 entity embeddings was computed using one of the three similarity metrics. The entities were then sorted based on their similarity scores, and the top k entities were retrieved for the reranking step.

Parameter	**Value**
BERT word embeddings	(105882, 768)
Additional special tokens	[Ms], [Me], [ENT]
Pooling function	Special Avg
Backpropagation loss	Contrastive loss
Mention/entity embeddings	128
Mention context window	5
Num epochs	5
Batch size	8
Learning rate	1e-5
Optimizer	Adam

Table 5.7: Model architecture and training parameters for the Bi-encoder model

Document-level Evaluation

The document-level evaluation pertains to a real-world scenario encountered in the initial dataset, where Electronic Health Records featured document-level labels of ICD-10 codes but lacked mention-level labels. This evaluation seeks to shed light on the models' capacity to

accurately assign document-level labels to the individual mentions. In this assessment, the model is presented with a mention, and for each mention, it considers only the labels that appear in the document to which the mention belongs.

Both the classifier and the bi-encoder are subjected to independent evaluations, focusing on their ability to predict which document-level label corresponds to a given mention. This scenario reflects the challenges posed by the absence of mention-level labels in the original EHRs and underscores the models' effectiveness in assigning appropriate document-level classifications to individual mentions.

Chapter 6

Results and Discussion

This section presents the performance evaluation of the hierarchical classifier and bi-encoder models. Through various configurations, the models' effectiveness in navigating the complex ICD-10 hierarchy is assessed. Results and discussions provide insights into strengths and potential areas for further exploration.

6.1 Evaluation of the Hierarchical model as a standalone classifier

The following subsection evaluates the hierarchical classifier as a standalone model, exploring its performance under different configurations. The aim is to understand how well the model is able to capture hierarchical relationships within the ICD-10 taxonomy and to identify optimal settings for effective medical code predictions. The models tested and compared against each other incude:

- Base classifier with no mention context
- Context Aware Base Classifier with CLS pooling
- Context Aware Base Classifier with special token average pooling
- Hierarchical Classifier with no mention context
- Context Aware Hierarchical Classifier with special token average pooling

The results of the evaluation tests are shown in table 6.1. The precision, recall, and f1 scores are calculated using macro averaging. In the initial phase of the assessments, three distinct configurations of the base classifier were scrutinized. These include the Base classifier without mention context, and two variants of the Context-Aware Base Classifier: one employing CLS token pooling for mention representations and the other utilizing the average of special tokens. Notably, the context-aware classifier exhibited superior performance

Model		Accuracy	Precision	Recall	F1
No context Base cls		78.63%	46.55%	49.12%	46.40%
Context-aware Base cls CLS pooling		0.64%	0.28%	0.67%	0.38%
Context-aware Base cls avg pooling		*80.69%*	*54.84%*	*56.18%*	*54.33%*
No context Hierarchical cls	(child)	79.02%	52.82%	54.84%	51.53%
	(parent)	93.60%	75.28%	72.48%	72.57%
Context-aware Hierarchical cls	(child)	**81.98%**	**60.65%**	**62.92%**	**60.34%**
	(parent)	95.11%	72.22%	69.83%	70.17%

Table 6.1: Hierarchical Classifier Evaluation Results

compared to the no-context base classifier, achieving an increase of *2.06%* in accuracy. Additionally, significant improvements of *8.29%*, *7.06%*, and *7.93%* were observed in precision, recall, and F1 scores, respectively. The CLS token classifier, however, displayed poor performance, yielding below 1% for all metrics and was consequently excluded from the subsequent hierarchical classification tests.

Upon comparing the hierarchical classifiers, it is evident that the context-aware model surpasses the context-unaware model across all metrics for child classes. Notably, the context-aware model demonstrates improvements of *2.78%*, *7.83%*, *8.08%*, and *8.81%* in accuracy, precision, recall, and F1 score, respectively. This observed trend aligns with the findings from the context-aware and non-context-aware base classifiers. The consistent enhancement in performance of context-aware models, when the appropriate pooling function is applied, suggests a slight increase in accuracy and a substantial improvement in precision, recall, and F1 scores. An intriguing observation in these results is that, for the parent classes, the precision, recall, and F1 scores exhibit a slight decrease for the context-aware model when compared with the context-unaware model. This phenomenon might suggest that overfitting for the parent classes might not have been entirely avoided.

In a final comparison, it is noteworthy that the context-aware base classifier outperforms the context-unaware hierarchical classifier, underscoring the perceived significance of context over hierarchy. However, the context-aware hierarchical model achieves the highest performance, emphasizing that the optimal approach lies in the synergistic combination of both contextual information and hierarchical structure.

In the subsequent section, the top-performing classifier model, namely the context-aware hierarchical classifier, will take on the role of conducting the reranking step to evaluate the

performance of the bi-encoder. This reranking process is used for refining the candidate concepts generated by the bi-encoder. In addition to that, the bi-encoder will be evaluated independently.

6.2 Evaluation of the Bi-Encoder

The evaluation of the bi-encoder was conducted in conjunction with the hierarchy classifier, leveraging various similarity measures to retrieve the top-k instances. This comprehensive assessment aimed to gauge the effectiveness of the bi-encoder in predicting labels within the hierarchical framework. Different similarity metrics were employed to explore the model's performance, offering insights into its capability to capture meaningful relationships between mentions and entities. For the retrieval of the candidate concepts, the similarity metrics evaluated were: dot product, cosine similarity, and euclidean distance.

For the comprehensive evaluation of each similarity metric, different candidate set sizes were tested, and the results are summarized in the set of tables 6.2. For context, the standalone performance of the hierarchical classifier is included as a reference point. Additionally, the bi-encoder underwent evaluation as a standalone classifier, wherein only the top 1 entity was retrieved and utilized as the predicted label. This analysis aims to provide insights into the bi-encoder's performance without the influence of a reranking step, offering a glimpse into its capabilities when tasked with singular label predictions.

The results indicate that while the bi-encoder didn't enhance the hierarchical classifier's overall performance, it demonstrated proficiency in retrieving correct labels, even within its top 5 candidates. Across all three metrics, an accuracy level of approximately 80% was achieved for candidate set sizes larger or equal to 5. Furthermore, each metric reached the maximum accuracy observed in the standalone hierarchical classifier, reaching 81.98%, with a set size limited to the top 60 candidate concepts.

For precision, recall, and F1 scores, the performance consistently hovered slightly below the baseline of the hierarchical model. Notably, even with smaller set sizes, the metrics demonstrated impressive performance. The dot product metric showcased the highest performance for a set size of 5, achieving an 81.20% accuracy, 57.67% precision, 60.93% recall, and 57.69% F1 score. Conversely, the euclidean distance metric exhibited the lowest scores for this set size, indicating a notable decline in performance. It's worth highlighting that nearly optimal performance was observed across all metrics for a set size of 50 candidates.

A noteworthy aspect of this analysis lies in the impressive performance of the bi-encoder when utilizing the dot product metric to retrieve the top 1 candidate concept. This suggests

Dot product

Top K	Accuracy	Precision	Recall	F1
100	81.98%	60.58%	62.92%	60.30%
60	81.98%	60.13%	62.43%	59.84%
50	81.85%	59.85%	61.65%	59.45%
10	81.33%	57.57%	60.85%	57.65%
5	81.20%	57.67%	60.93%	57.69%
1	70.78%	46.65%	48.14%	44.43%

Cosine Similarity

Top K	Accuracy	Precision	Recall	F1
100	81.98%	60.17%	62.24%	59.86%
60	81.98%	59.80%	62.92%	59.88%
50	81.85%	59.85%	61.65%	59.45%
10	81.33%	55.96%	59.43%	56.31%
5	80.95%	56.46%	60.45%	56.85%
1	68.85%	45.70%	47.30%	43.64%

Euclidean Distance

Top K	Accuracy	Precision	Recall	F1
100	81.98%	59.82%	62.43%	59.60%
60	81.98%	59.38%	62.83%	59.94%
50	81.85%	59.14%	62.04%	59.14%
10	80.43%	56.17%	59.89%	56.28%
5	78.63%	52.86%	56.25%	52.83%
1	65.52%	45.69%	44.34%	41.33%

Standalone Hierarchical	**81.98%**	**60.65%**	**62.92%**	**60.34%**

Table 6.2: Evaluation of two-stage EL for various numbers of candidates

that the bi-encoder effectively learned the similarities and dissimilarities among mention-entity pairs, successfully retrieving the correct entity corresponding to a given mention, even when tasked with obtaining the single most similar entity embedding. Such results bode well for the prospective application of bi-encoders in the entity-linking task within the Greek medical domain.

In summary, the bi-encoder displayed significant promise in accurately retrieving mention-entity pairs, even within very small candidate sets, showcasing optimal outcomes for a set of 50 candidates across all assessed metrics. The bi-encoder's performance seems to be limited from the reranking step of the hierarchical classifier, which has set an upper limit at 81.98%,

60.65%, 62.92%, and 60.34% for accuracy, precision, recall, and F1 score, respectively.

It was deemed purposeful to continue the evaluation without the influence of the hierarchical classifier. For that, the bi-encoder was evaluated for its ability to retrieve the correct label within its top-k candidates. It is evident by the results that the bi-encoder is capable of retrieving the correct label in almost 99% of the candidate sets, within the top 50 retrieved candidates, making it a powerful tool for the candidate generation step. Notably, for the top 5 candidates, representing only 2.5% of the entire label set, the bi-encoder attains an impressive 94.72% accuracy of containing the correct label within the retrieved set. This suggests that the bi-encoder excels in providing highly relevant and accurate candidate sets, even with the inclusion of unknown labels, demonstrating its effectiveness in addressing the complexities of entity linking tasks. The results of all k tested are shown in table 6.3.

Top k	**k as a percentage of the labelset**	**Percentage of Candidate Sets that contain the correct label**
5	2.5%	94.72%
10	5%	96.65%
20	10%	97.68%
35	17.5%	98.32%
50	25%	98.84%
60	30%	98.97%
100	50%	99.48%

Table 6.3: Bi Encoder's ability to retrieve the correct label within the top-k candidates

Upon further investigation into the relationship between the number of retrieved candidates (k) and the accuracy of the candidate set, a distinct pattern emerged. The rate of accuracy increase gradually decelerated with increasing k, revealing a logarithmic growth. This logarithmic trend signifies that while accuracy improves with additional candidates, the rate of improvement diminishes. The pattern is clearly shown in figure 6.1. The practical implication is the identification of an optimal k where including more candidates provides minimal accuracy improvement. This balance is crucial for resource efficiency, considering the computational cost and marginal gains in accuracy. It's worth noting that the optimal k identified through limited testing on a small set may be smaller than the k needed for real-world applications.

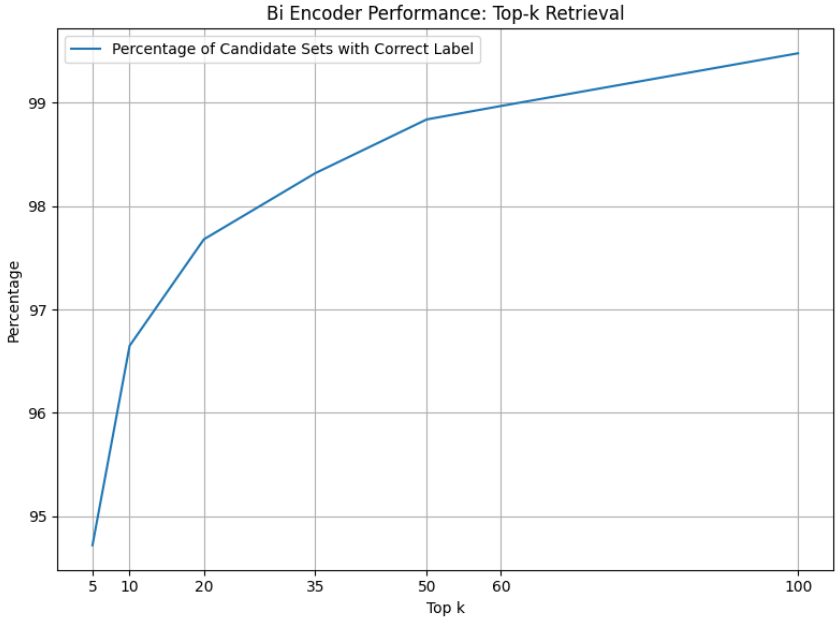

Figure 6.1: Relationship Between the Number of k and Candidate Set Accuracy in Bi-Encoder Evaluation

6.3 Evaluation with document-level labels

The original corpora, prior to undergoing the annotation process by medical professionals, featured document-level labels. In this context, each discharge letter in the corpora had associated ICD-10 codes, which served as labels for the entire document. The purpose of this evaluation is to assess the performance of the proposed models when presented with document-level labels.

As discussed in the dataset section, the dataset was split into train, validation, and test sets. Those splits were performed disregarding the document each mention was in. As a result, each set comprises segmented documents. The extraction of document-level labels was carried out for entire documents, rather than solely for the mentions incorporated in the test set.

To gain a comprehensive insight into the models' performance, a closer examination of the document-level labels is essential. In the previously discussed test set comprising 777 mentions, the mean value of labels per document stands at 8.77. The distribution of the number of labels spans from a minimum of 1 to a maximum of 17. Notably, only 25% of the samples exhibit labels below 6, 50% fall below 8, and 75% register below 11. This distribution offers a nuanced perspective on the frequency and variability of labels within the dataset. A more thorough analysis of the document level labels distribution is shown in the histogram in Figure 6.2.

The hierarchical classifier and the bi-encoder were subjected to testing using the document-

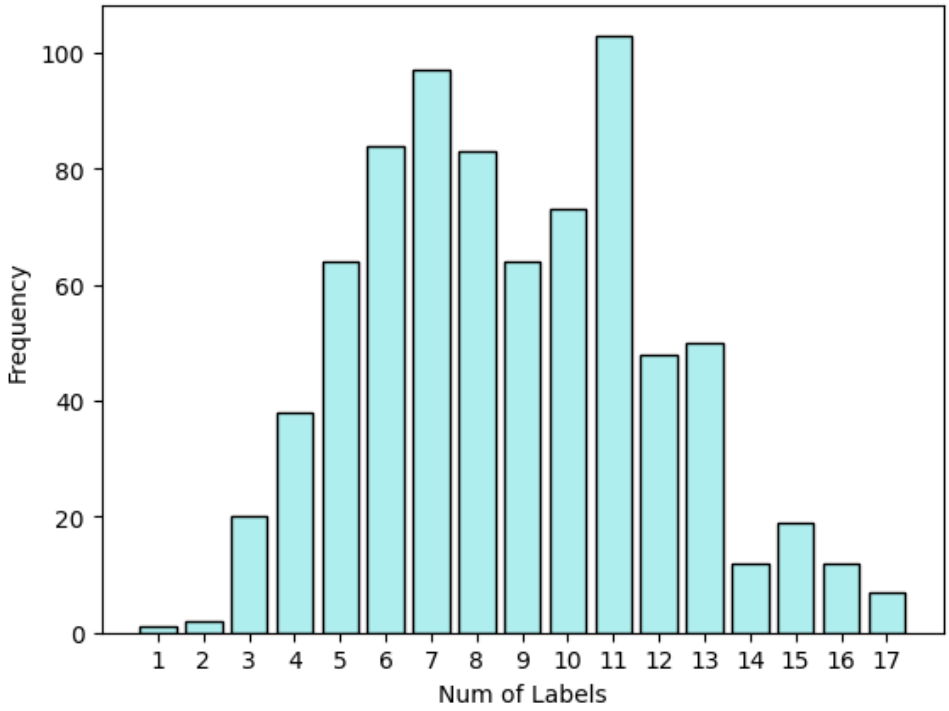

Figure 6.2: Distribution of the number of document-level labels

level label set as the only candidate labels. In the case of the hierarchical classifier, document-level labels underwent reranking based on the probabilities from the softmax layer, with the top value selected as the predicted label.

Similarly, the bi-encoder was employed to predict the correct value by choosing the label from the document-level labels whose embedding more closely matched the embedding of the mention. For evaluation purposes, the best-performing classifier, namely the context-aware hierarchical classifier with average special pooling, was selected. Additionally, the chosen similarity metric for the bi-encoder evaluation was the one that demonstrated superior performance when evaluated as a standalone classifier, specifically the dot product.

As expected, the performance of both models exhibited a significant improvement, surpassing 10% across all metrics, as the utilization of document-level labels aided in filtering out a substantial portion of false classes. Notably, the bi-encoder demonstrated a higher ability to predict the correct label compared to the classifier. It achieved the highest accuracy so far, reaching 93.82%, surpassing the classifier by 1.01%. Furthermore, the bi-encoder outperformed the classifier by almost 10% in precision, recall, and F1 scores, showcasing its superior effectiveness in this document-level label scenario.

An additional evaluation was conducted to assess the bi-encoder's ability to retrieve the correct labels within a top-k set. Notably, the investigation revealed that the correct label was consistently retrieved within the top 6 candidates with 100% accuracy, despite the labelset

reaching up to 17 labels in number. Furthermore, metrics were computed for the top-2, top-3 and top-5, operating under the assumption that the correct label was successfully predicted if it appeared within these top k retrieved labels. The results of all the aforementioned evaluations are summarized in table 6.4.

Model		Accuracy	Precision	Recall	F1
Hierarchical cls		92.79%	73.64%	73.90%	73.00%
Bi encoder	top-1	93.82%	82.21%	82.95%	81.79%
	top-2	97.68%	91.35%	92.16%	91.34%
	top-3	99.09%	97.30%	97.63%	97.31%
	top-5	99.61%	99.49%	99.49%	99.38%

Table 6.4: Model Evaluation for document-level labels

6.3.1 Analysis of Unpredicted Labels

Despite achieving a top accuracy of 93.82%, the models were still unable to perform correct classifications for some labels. This section conducts a detailed analysis of labels that presented challenges for the models, examining instances where both the context-aware hierarchical classifier and the bi-encoder faced difficulty in predicting accurate labels. The granular examination of these unpredicted labels aims to uncover patterns or specific characteristics contributing to the challenges in prediction.

Hierarchical Classifier Analysis

The first thing to note for the hierarchical classifier is that since it was only trained upon the labels of the training set, despite having knowledge of the unseen labels present in the test set, it is unable to predict any of them. That behavior was expected since no associations were made for the specific labels. In contrast, the subsequent examination of the bi-encoder, detailed in the following paragraph, reveals that it has a greater ability for generalization since it was seen to effectively create embeddings for descriptions and mentions not seen in the training set.

An in-depth analysis was undertaken to investigate instances where the classifier failed to make accurate predictions. A comprehensive summary of this analysis is presented in Figure 6.3. Furthermore, to gain deeper insights into these misclassifications, a focused exploration was conducted on the most frequently occurring erroneous predictions. The detailed breakdown of these instances, including the associated mentions and predicted labels, is provided in Table 6.5.

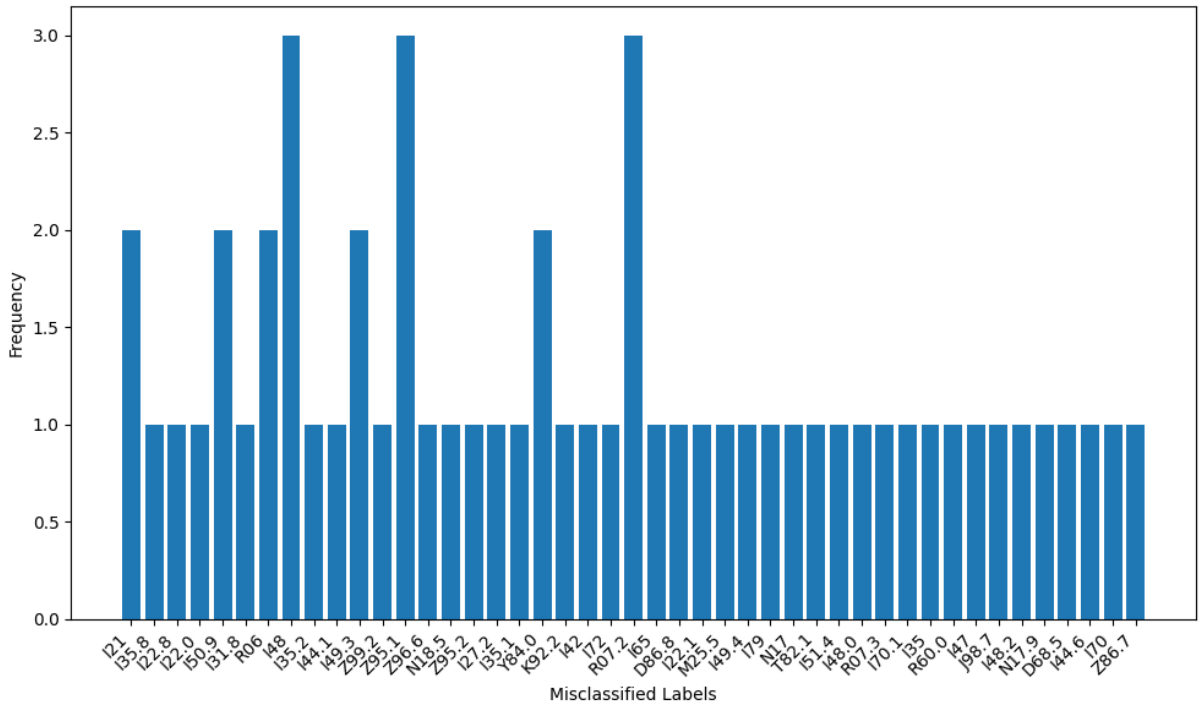

Figure 6.3: Hierarchical Cls Label Misclassification Distribution

True Label	Mention Text	Mispredicted Labels
I21	‘κατασπαση st’, ‘non stemi’	I22.0, I21.4
I48	‘παροξυσμικος κολπικος πτερυγισμος’, ‘κμ’, ‘af αγνωστου εναρξεως’	I21.4, I48.9
Z95.1	‘αορτοστεφανιαια παρακαμψη’	Z95.5, Y84.0
R07.2	‘προκαρδιου αλγους’, ‘οπισθοστερνικο αλγος’, ‘συσφιγκτικου αλγους στο στερνο’	R07

Table 6.5: Hierarchical cls’ most frequent mispredicted labels and corresponding mentions

The result analysis reveals a pattern in mispredictions by the Hierarchical Classifier. For the true label I21, mentions like ’κατασπαση st’ and ’non stemi’ were mispredicted as I22.0 and I21.4, suggesting potential challenges in distinguishing specific subtypes within this overarching category. However, it’s noteworthy that despite these mispredictions, the classifier demonstrated an ability to discern the broader category behind the codes. In several instances, it successfully classified adjacent codes, such as I48 being classified as I48.9 and R07.2 being classified as just R07. This proficiency in capturing the semantic context is further supported by the classifier’s superior performance in block classification. These findings underscore the

model's capacity to navigate the intricacies of medical coding, providing valuable insights for refining and enhancing its performance in future iterations.

Bi encoder analysis for the top-1 retrieved candidate

The analysis of the bi-encoder's predictions for the top-1 reveals instances of label misclassifications within similar code blocks and even the same base code. For example, the mention "avf" originally labeled as I22.0 is predicted as I22.8, and "επασβεστωση αορτικης βαλβιδας με καλη διανοιξη και ηπια στενωση" labeled as I35.0 is predicted as I35.8. Additionally, there are cases where mentions from different codes in the same block are misclassified, such as the label I21 being predicted as I22.0 for "κατασπαση st" and I25.2 being predicted as I21.9 for "παλαιο οεμ." Another category of misclassifications involves predictions as labels from entirely different categories. For instance, "επιπωματισμου," belonging to I31.8, is predicted as R60.0, and "στεφανιογραφιας" is misclassified as Z95.5 while the correct label is Y84.0. The label misclassification distribution is shown in Figure 6.4, while the most frequent misclassified labels are further analyzed in Table 6.6

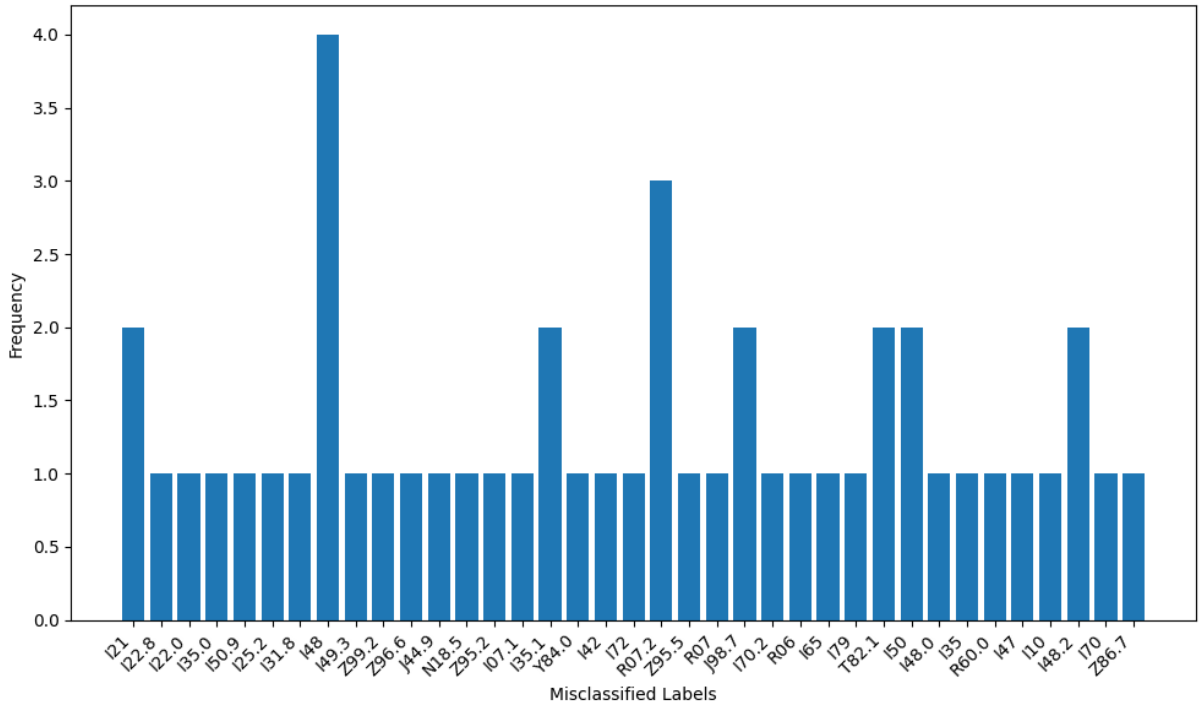

Figure 6.4: Bi-encoder Label Misclassification Distribution

The first thought is to see if the misclassified labels actually belonged in the training set, to asses how many instances the model hasn't seen during training. Out of the 48 misclassified instances, 41 were found to be present in the training set, indicating that the majority of misclassifications occurred for labels the model had encountered during training. Only 7

True Label	Mention Text	Mispredicted Labels
I21	‘κατασπαση st’, ‘non stemi’	I21.9, E11.8
I48	‘παροξυσμικος κολπικος πτερυγισμος’, ‘κμ’, ‘κμ’, ‘κολπικη μαρμαρυγη’	I21.4, Z95.1, I48.0
I35.1	‘αor’, ‘avr’	Z95.5
R07.2	‘προκαρδιου αλγους’, ‘οπισθοστερνικο αλγος’, ‘συσφιγκτικου αλγους στο στερνο’	I10, I20.0
I50	‘καρδιακη ανεπαρκεια με μη διατηρημενο κλασμα εξωθησης’, ‘απορρυθμιση καρδιακης ανεπαρκειας’	N18, I25.1
I48.2	‘xkm’, ‘κχμ’	I07.1, Z95.0

Table 6.6: Bi-encoder’s most frequent mispredicted labels and corresponding mentions

instances were misclassified due to the model’s lack of exposure to them during training. At this point, it is also worth examining if the model managed to correctly classify any labels that were not present in the training set. Remarkably, the bi-encoder managed to correctly extract embeddings for 8 different descriptions not seen in the training set and successfully associate them with corresponding mention embeddings, with their corresponding labels being ‘I35.8,’ ‘D86.8,’ ‘M25.5,’ ‘N17,’ ‘I51.4,’ ‘I70.1,’ ‘D68.5,’ and ‘I44.6.’ This discovery sheds light on the model’s ability to generalize and accurately classify instances that were not explicitly encountered during its training phase.

Upon comparing Tables 6.5 and 6.6, notable distinctions and resemblances emerge in the misclassification patterns of the hierarchical classifier and the bi-encoder models. Notably, both models exhibit challenges in accurately predicting instances of the I21 and I48 codes. For instance, mentions associated with code I48 present variations, including the abbreviation ‘κμ′ for ‘κολπική μαρμαρυγή’, which appears to perplex both models. Additionally, the misclassification of adjacent codes, such as I48.2, suggests a common struggle in capturing subtle nuances of the similar abbreviations. Similarly, the misprediction of code R07.2, characterized by varying mentions, underscores the complexity of accurately categorizing certain codes. Interestingly, while the hierarchical classifier tends to favor predicting code R07 over R07.2, likely due to its higher frequency in the dataset being amongst the top 50 most frequent codes, the bi-encoder model exhibits a preference for codes within the ‘I’ chapter of

the ICD-10 classification.

Bi encoder analysis for the top-5 retrieved candidates

In the scenario involving the top-5 candidates retrieved by the bi-encoder, it is noteworthy that the true label was absent in the set for only 3 of the tested mentions. Significantly, all three of these instances were included in the training set. This implies that the model adeptly extracted embeddings for all the unseen cases and accurately linked them with the respective tested labels within the top-5 candidates. Interestingly, the three instances not contained in the top-5 were consistently positioned in the 6th place in terms of similarity metrics. The analysis is summarized on table 6.7.

Bi-Encoder

Analysis Type	**Observations for top-1**	**Observations for top-5**
Predictions	Instances of label misclassifications within similar code blocks and the same base code. Misclassifications involving predictions as labels from entirely different categories were also observed.	Succussfully retrieved almost all correct entities within the top-5 candidates.
Presence in Training Set	Out of 48 misclassified labels, 41 were present in the training set. Only 7 instances were misclassified due to the model's lack of exposure to them during training.	All labels that were not retrieved within the top-5 were seen in the training set.
Generalization to Unseen Labels	The bi-encoder accurately associated embeddings for 8 different descriptions not seen in the training set, with their corresponding mention embeddings.	The bi-encoder showed great generalizability within the top-5 candidates, having correctly retrieved embeddings of all 15 unseen cases.

Table 6.7: Bi-Encoder Analysis Summary

Chapter 7

Conclusions and Future Work

The pursuit of efficient and accurate entity linking (EL) methodologies continues to be paramount in natural language processing (NLP), particularly within specialized domains such as medical literature. This study delved into the intricate landscape of EL, focusing on Greek medical texts and utilizing the International Classification of Diseases, 10th Revision (ICD-10) as the knowledge base. By amalgamating contemporary methodologies and innovative approaches, this research endeavors to contribute nuanced insights into the evolving strategies of entity linking, shedding light on optimal configurations and novel paradigms for tackling EL challenges. In this section, the key findings and implications of this study are presented, followed by a discussion of its limitations and avenues for future research.

7.1 Conclusions

The entity linking landscape has witnessed evolving strategies, and this study contributes to this dynamic field by adopting a nuances methodology inspired by contemporary literature. Drawing upon this foundation, the approach embraces two distinctive paradigms for entity linking tasks, each tailored to address specific challenges encountered in the context of Greek medical texts and the ICD-10. The first approach aligns with the one-staged EL modeled as a text classification task, employing a standalone classifier. Building on the natural hierarchical structure of the ICD-10, a model is introduced that utilizes that hierarchy by considering both codes and their parent blocks in the training and classification process.

The second approach navigates the complexity of entity linking through a two-stage process involving candidate concept generation and concept reranking. In a departure from the conventional use of cross-encoders for the candidate reranking step, the methodology combines the robustness of the bi-encoder architecture for candidate generation with the proposed classifier, offering a novel perspective on the two-stage entity linking paradigm.

7.1.1 Key Findings and Implications

This section outlines the crucial findings obtained through the application of the proposed methodologies. These insights shed light on the performance of various models in the task of entity linking in Greek medical texts.

From the initial series of tests aimed at determining the optimal classifier configuration, several key observations emerged:

- Context-aware classifiers consistently outperformed their non-context-aware counterparts.
- Classifiers incorporating the ICD-10 hierarchy demonstrated superior performance compared to their counterparts lacking hierarchy.
- The highest-performing classifier was the hierarchical context-aware model. This emphasized the importance of considering both context and hierarchy, showcasing a significant improvement: a 1.29%, 5.81%, 6.74%, and 6.01% increase in accuracy, precision, recall, and F1 scores, respectively, over the second-best classifier, which was the context-aware non-hierarchical model.

The bi-encoder model demonstrated remarkable proficiency in the candidate retrieval step, even with limited training data. Two evaluation settings were considered, and are described below along with their key findings:

1. Bi-Encoder with Hierarchical Classifier for Reranking: In conjunction with the hierarchical classifier performing the candidate generation step. The bi-encoder was evaluated under three similarity metrics: dot product, cosine similarity, and Euclidean distance. Key findings include:
    - All metrics yielded optimal results for the top-50 candidates, with the dot product demonstrating superior performance for sets with fewer than 10 candidates.
    - Including the bi-encoder for the candidate generation step did not improve the overall performance of the final classification of the hierarchical classifier.
    - Despite not improving the performance, the candidate generation step was good enough such as to yield optimal results for as low as 50 candidates, and keep the results satisfyingly high even with the top 5 candidates with 81.20% 57.67% 60.93% 57.69% for accuracy, precision, recall and F1 scores respectively

2. Bi-Encoder without Reranking: Evaluated independently for its ability to retrieve the correct label within the top-k candidate concepts. For this evaluation, the similarity metric opted for was the dot product due to its superior performance. Key findings include:

    - The model achieved an impressive 94.72% accuracy in retrieving the correct label within the top-5 candidates, suggesting promising results with a more robust model for the reranking step.
    - When acting as a classifier (top-1 retrieval), the model achieved a score of 70.78%, 46.65%, 48.14%, and 44.43% for accuracy, precision, recall, and F1 scores, respectively.
    - The analysis of accuracy increase with varying k revealed a logarithmic growth, indicating diminishing returns with additional candidates and emphasizing the importance of identifying an optimal k for efficiency.

At the project's inception, the discharge documents that formed the basis of this study solely contained document-level labels. Drawing inspiration from this real-life scenario, an evaluation was conducted specifically focusing on these document-level labels. In this evaluation setting, document-level labels served as the exclusive candidate concepts. Both models underwent assessment in this context, yielding the following results:

1. Hierarchical Classifier:

    - Achieved an accuracy of 92.79%, with precision, recall, and F1 scores of 73.64%, 73.90%, and 73.00%, respectively.
    - Misclassifications often occurred with codes adjacent to the true label or from the same block. Other errors were attributed to unseen labels during training.

2. Bi-encoder:

    - Acting as a classifier (top-1), the bi-encoder achieved an impressive accuracy of 93.82%, with precision, recall, and F1 scores of 82.21%, 82.95%, and 81.79%, respectively.
    - Showcasing promising results for unseen labels, the bi-encoder demonstrated notable generalization by accurately predicting labels not included in the training set.

- Substantially improved performance when evaluated for the top-2 (97.68%, 91.35%, 92.16%, 91.34%) and reached a remarkable 99% accuracy for the top-3 concepts.
- Despite an average of 8.7 labels per document (max: 17), the bi-encoder's remarkable accuracy in top-ranked predictions, especially for the top 2 (or 3), highlights its effectiveness in real-world scenarios where numerous candidates are involved.

In conclusion, the findings validate the viability of employing BERT-based models for the nuanced nature of Greek medical texts. The introduction of a hierarchical element in the classifier demonstrates a noticeable improvement in performance compared to models without hierarchy. However, it is evident that the current model still faces challenges, and further enhancements are imperative for optimal results, as discussed in the subsequent Future Work section. Notably, the bi-encoder showcases remarkable efficacy in the candidate generation step, successfully capturing the intricacies embedded in medical text and the nuanced descriptors within the ICD-10. While in limited testing, the model also shows promising results indicating its potential for good generalizability. These results not only underscore its significance in advancing entity linking tasks within the medical domain but also suggest promising avenues for leveraging the bi-encoder model to address document-level label tasks. The versatility and favorable outcomes of the bi-encoder model pave the way for broader applications, showcasing its potential impact and relevance in various domains.

7.1.2 Limitations

While this study has revealed significant insights and delivered promising results for the task of entity linking in the context of Greek medical texts, it is essential to acknowledge several limitations. Firstly, the evaluation dataset, though representative of Greek medical discharge documents collected from hospitals nationwide, may not fully encapsulate all linguistic nuances and variations present in real-world applications. This limitation arises due to the dataset being a limited sample of documents from a specific medical ward. Additionally, the training data used to develop and fine-tune the models is limited in scope, influenced by the small number of documents in the dataset and the even smaller subset of training labels used for annotations. Unfortunately, additional test data was not made available to assess the generalization abilities of the bi-encoder to retrieve labels not present in the training set. Lastly, computational resources and infrastructure constraints in this research limited the implementation of techniques such as pre-training of BERT embeddings on Greek medical texts and allowing it to better understand the contextual nuances of this particular domain-specific task.

7.2 Future Work

While this study has provided valuable insights into implementing entity linking methodologies for Greek medical texts, there remain several avenues for future exploration and improvement. The findings of this research have illuminated potential areas for refinement and expansion, laying the groundwork for future endeavors in the realm of entity linking. The following are key directions for future work that can further advance the field and address the identified limitations.

In future iterations, the hierarchical classifier can be enhanced through several components. One avenue for improvement involves refining the hierarchical loss function to potentially include more hierarchical information. This could entail incorporating a larger portion of the ICD-10 hierarchy, allowing the model to better capture hierarchical relationships more comprehensively. Additionally, the introduction of a strict hierarchical prediction mechanism could be explored, where child and parent codes follow a hierarchical monotonicity. Such an approach not only has the potential to contribute to improved accuracy but also enhances model interpretability by revealing the hierarchical path the model followed to reach the final code prediction.

Given that the models were designed for entity linking in a low-resource language and low-resource domain, several strategies could be explored to overcome this limitation. One approach involves investigating additional pre-training of BERT models on Greek EHRs or other Greek medical texts. This step would enable BERT embeddings to better capture the nuances of the medical domain in Greek, potentially resulting in improved model performance. Another avenue is the incorporation of models like XLM-R, which have demonstrated proficiency in multilingual tasks. This could enhance the model's ability to handle the complexities of a low-resource language by leveraging knowledge from other languages. However, it's crucial to acknowledge the challenges associated with Electronic Health Records, which often contain numerous abbreviations and doctor jargon. Techniques such as supervision from resource-rich languages and translation models may be challenging to implement effectively in this context.

A key aspect for future work involves the enhancement of training data. Expanding the scope and diversity of the training dataset could significantly improve the models' generalization capabilities. It would be an interesting study to explore the capabilities of the models once the complete dataset of 1K documents is available. This expanded dataset could expose the models to a wider range of linguistic nuances and variations present in real-world applications, contributing to improved robustness. In addition to this, an important aspect is

a broader evaluation. Currently, the evaluation has been conducted on a representative but limited set of medical discharge documents. To better assess the generalizability of the bi-encoder model in the specific task, it is imperative to expand the evaluation to a larger scale. This entails encompassing a broader set of medical documents and ICD-10 labels, simulating a more comprehensive and diverse real-world scenario. A broader evaluation will provide a more nuanced understanding of the models' performance across varied contexts and facilitate the identification of potential challenges and opportunities for improvement.

In conclusion, the future work for this study holds promising avenues for advancing the field of entity linking within the domain of Greek medical texts and the ICD-10. The proposed enhancements to the hierarchical classifier, including refining the loss function and exploring stricter hierarchical prediction, offer opportunities to improve model interpretability and performance. Additionally, the incorporation of techniques to address the challenges of working with a low-resource language and specialized medical domain would further improve the models' robustness. Furthermore, expanding the training dataset enhances the models' understanding of Greek medical nuances. A broader evaluation with diverse medical documents and ICD-10 labels is crucial for refining the bi-encoder model's effectiveness in entity linking tasks within the Greek medical domain.

Bibliography

[1] N. Sager, Carol Friedman, E. Chi, C. Macleod, S. Chen, and S. Johnson. The analysis and processing of clinical narrative. *Medinfo 86*, pages 1101–1105, 1986.

[2] Carol Friedman. Towards a comprehensive medical language processing system: methods and issues. *Proc AMIA Annu Fall Symp*, pages 595–599, 1997.

[3] Alan R. Aronson, Olivier Bodenreider, Hsinchun Francis Chang, Susanne M. Humphrey, James G. Mork, Stuart J. Nelson, Thomas C. Rindflesch, and W. John Wilbur. The nlm indexing initiative. *Proc AMIA Symp*, pages 17–21, 2000.

[4] Daisuke Okanohara, Yusuke Miyao, Yoshimasa Tsuruoka, and Jun'ichi Tsujii. Improving the scalability of semi-markov conditional random fields for named entity recognition. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, page 465–472, USA, 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220234. URL https://doi.org/10.3115/1220175.1220234.

[5] Yulan Lu, Donghong Ji, and Xuan Yao. Chemdner system with mixed conditional random fields and multi-scale word clustering. *J Cheminform*, 7(Suppl 1):S4, 2015. doi: 10.1186/1758-2946-7-S1-S4. URL https://doi.org/10.1186/1758-2946-7-S1-S4.

[6] Hong Li, Qiaozhu Chen, Buzhou Tang, and et al. Cnn-based ranking for biomedical entity normalization. *BMC Bioinformatics*, 18(Suppl 11):385, 2017. doi: 10.1186/s12859-017-1805-7. URL https://doi.org/10.1186/s12859-017-1805-7.

[7] Yufeng Luo, Weiyi Sun, and Anna Rumshisky. A hybrid normalization method for medical concepts in clinical narrative using semantic matching. *AMIA Jt Summits Transl Sci Proc*, pages 732–740, 2019.

[8] Greg Durrett and Dan Klein. A joint model for entity analysis: Coreference, typing, and linking. *Transactions of the Association for Computational Linguistics*, 2:477–490, 2014. doi: 10.1162/tacl_a_00197. URL https://doi.org/10.1162/tacl_a_00197.

[9] Robert Leaman, Rezarta Islamaj Doğan, and Zhiyong Lu. Dnorm: disease name normalization with pairwise learning to rank. *Bioinformatics*, 29(22):2909–2917, 2013. doi: 10.1093/bioinformatics/btt474. URL https://doi.org/10.1093/bioinformatics/btt474.

[10] Peng-Hsuan Li, Ruo-Ping Dong, Yu-Siang Wang, Ju-Chieh Chou, and Wei-Yun Ma. Leveraging linguistic structures for named entity recognition with bidirectional recursive neural networks. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2664–2669, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1282. URL https://aclanthology.org/D17-1282.

[11] Emma Strubell, Patrick Verga, David Belanger, and Andrew McCallum. Fast and accurate entity recognition with iterated dilated convolutions, 2017.

[12] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[13] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations, 2018.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[16] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training, 2018. URL https://api.semanticscholar.org/CorpusID:49313245.

[17] Alexander Nesterov and Dmitry Umerenkov. Distantly supervised end-to-end medical entity extraction from electronic health records with human-level quality, 2022.

[18] Luka Gligic, Andrey Kormilitzin, Paul Goldberg, and Alejo Nevado-Holgado. Named entity recognition in electronic health records using transfer learning bootstrapped neural networks. *Neural Networks*, 121:132–139, 2020. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2019.08.032. URL https://www.sciencedirect.com/science/article/pii/S089360801930259X.

[19] Mahanazuddin Syed, Shaymaa Al-Shukri, Shorabuddin Syed, Kevin Sexton, Melody L. Greer, Meredith Zozus, Sudeepa Bhattacharyya, and Fred Prior. Deidner corpus:

Annotation of clinical discharge summary notes for named entity recognition using brat tool. *Studies in Health Technology and Informatics*, 281:432–436, 2021. doi: 10.3233/SHTI210195.

[20] Jui-Shan Chen, Wei-Chih Lin, Shuo Yang, Michael F. Chiang, and Michelle R. Hribar. Development of an open-source annotated glaucoma medication dataset from clinical notes in the electronic health record. *Transl Vis Sci Technol*, 11(11):20, 2022. doi: 10.1167/tvst.11.11.20.

[21] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. brat: A web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, 2012.

[22] Pinal Patel, Disha Davey, Vishal Panchal, and Parth Pathak. Annotation of a large clinical entity corpus. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2033–2042, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1228. URL `https://aclanthology.org/D18-1228`.

[23] Lucas E. S. e Oliveira, Anselmo C. Peters, Ana M. P. da Silva, Caio P. Gebeluca, Yohan B. Gumiel, Luis M. M. Cintho, Diego R. Carvalho, Sami Al Hasan, and Claudia M. C. Moro. Semclinbr - a multi-institutional and multi-specialty semantically annotated corpus for portuguese clinical nlp tasks. *J Biomed Semantics*, 13(1):13, 2022. doi: 10.1186/s13326-022-00269-1.

[24] Fei Li, Yilong Jin, Wei Liu, Brijesh P. S. Rawat, Peipei Cai, and Hong Yu. Fine-tuning bidirectional encoder representations from transformers (bert)-based models on large-scale electronic health record notes: An empirical study. *JMIR Med Inform*, 7(3):e14830, 2019. doi: 10.2196/14830.

[25] Eileen Chang and Javed Mostafa. The use of snomed ct, 2013-2020: a literature review. *J Am Med Inform Assoc*, 28(9):2017–2026, 2021. doi: 10.1093/jamia/ocab084.

[26] Antonio Miranda-Escalada, Aitor Gonzalez-Agirre, Jordi Armengol-Estapé, and Martin Krallinger. Overview of automatic clinical coding: annotations, guidelines, and solutions for non-english clinical cases at codiesp track of clef ehealth 2020. In *Working Notes of Conference and Labs of the Evaluation (CLEF) Forum. CEUR Workshop Proceedings*, 2020.

[27] Paul L. Schuyler, William T. Hole, Mark S. Tuttle, and David D. Sherertz. The umls metathesaurus: representing different views of biomedical concepts. *Bull Med Libr Assoc*, 81(2):217–222, 1993.

[28] Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. Ncbi disease corpus: A resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, 47:1–10, 2014. ISSN 1532-0464. doi: 10.1016/j.jbi.2013.12.006. URL https://www.sciencedirect.com/science/article/pii/S1532046413001974.

[29] Sunil Mohan and Donghui Li. Medmentions: A large biomedical corpus annotated with umls concepts, 2019.

[30] Natalia Loukachevitch, Suresh Manandhar, Evgueni Baral, Ivan Rozhkov, Pavel Braslavski, Vladimir Ivanov, Tatiana Batura, and Elena Tutubalina. Nerel-bio: a dataset of biomedical abstracts annotated with nested named entities. *Bioinformatics*, 39(4): btad161, 2023. doi: 10.1093/bioinformatics/btad161.

[31] Leonardo Campillos, Louise Deléger, Cyril Grouin, Thierry Hamon, Anne-Laure Ligozat, and et al. A french clinical corpus with comprehensive semantic annotations: development of the medical entity and relation limsi annotated text corpus (merlot). *Language Resources and Evaluation*, 52(2):571–601, 2017. doi: 10.1007/s10579-017-9382-y. URL https://hal.archives-ouvertes.fr/hal-01631743.

[32] Ido Lerner, Nicolas Paris, and Xavier Tannier. Terminologies augmented recurrent neural network model for clinical named entity recognition. *J Biomed Inform*, 102:31837473, 2020. doi: 10.1016/j.jbi.2019.103356.

[33] Laura Campillos-Llanos, Alicia Valverde-Mateos, Ana Capllonch-Carrión, and Antonio Moreno-Sandoval. A clinical trials corpus annotated with umls entities to enhance the access to evidence-based medicine. *BMC Med Inform Decis Mak*, 21(1):69, 2021. doi: 10.1186/s12911-021-01395-z. Erratum in: BMC Med Inform Decis Mak. 2021 Apr 7;21(1):118.

[34] Antonio Miranda-Escalada, Luis Gasco, Salvador Lima-López, Eulàlia Farré-Maduell, Darryl Estrada, Anastasios Nentidis, Anastasia Krithara, Georgios Katsimpras, Georgios Paliouras, and Martin Krallinger. Overview of distemist at bioasq: Automatic detection and normalization of diseases from clinical texts: results, methods, evaluation and multilingual resources. In *Working Notes of Conference and Labs of the Evaluation (CLEF) Forum*.

CEUR Workshop Proceedings, 2022. URL https://ceur-ws.org/Vol-3180/paper-11.pdf.

[35] Salvador Lima-López, Eulàlia Farré-Maduell, Luis Gasco, Anastasios Nentidis, Anastasia Krithara, Georgios Katsimpras, Georgios Paliouras, and Martin Krallinger. Overview of medprocner task on medical procedure detection and entity linking at bioasq 2023. *Working Notes of CLEF*, 2023. URL https://ceur-ws.org/Vol-3497/paper-002.pdf.

[36] Özge Sevgili, Artem Shelmanov, Mikhail Arkhipov, Alexander Panchenko, and Chris Biemann. Neural entity linking: A survey of models based on deep learning. *Semantic Web*, art. semantic-web/sw222986, January 2022. doi: 10.3233/SW-222986. URL https://doi.org/10.3233/SW-222986.

[37] Samuel Broscheit. Investigating entity knowledge in bert with simple neural end-to-end entity linking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. Association for Computational Linguistics, 2019. doi: 10.18653/v1/k19-1063. URL http://dx.doi.org/10.18653/v1/K19-1063.

[38] Nina Poerner, Ulli Waltinger, and Hinrich Schütze. E-BERT: Efficient-yet-effective entity embeddings for BERT. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 803–818, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.71. URL https://aclanthology.org/2020.findings-emnlp.71.

[39] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive entity retrieval. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=5k8F6UU39V.

[40] Phong Le and Ivan Titov. Improving entity linking by modeling latent relations between mentions. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1595–1604, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1148. URL https://aclanthology.org/P18-1148.

[41] Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. Zero-shot entity linking by reading entity descriptions. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy,

July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1335. URL https://aclanthology.org/P19-1335.

[42] Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. Scalable zero-shot entity linking with dense entity retrieval. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.519. URL https://aclanthology.org/2020.emnlp-main.519.

[43] Feng Nie, Yunbo Cao, Jinpeng Wang, Chin-Yew Lin, and Rong Pan. Mention and entity description co-attention for entity disambiguation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. doi: 10.1609/aaai.v32i1.12043. URL https://ojs.aaai.org/index.php/AAAI/article/view/12043.

[44] Tang H. Sun X. Jin B. and Zhang. A bidirectional multi-paragraph reading model for zero-shot entity linking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13889–13897, May 2021. doi: 10.1609/aaai.v35i15.17636. URL https://ojs.aaai.org/index.php/AAAI/article/view/17636.

[45] Xingyu Fu, Weijia Shi, Xiaodong Yu, Zian Zhao, and Dan Roth. Design challenges in low-resource cross-lingual entity linking. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6418–6432, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.521. URL https://aclanthology.org/2020.emnlp-main.521.

[46] Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. Cross-lingual name tagging and linking for 282 languages. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1178. URL https://aclanthology.org/P17-1178.

[47] Chen-Tse Tsai and Dan Roth. Cross-lingual wikification using multilingual embeddings. In Kevin Knight, Ani Nenkova, and Owen Rambow, editors, *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 589–598, San Diego, California, June

2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1072. URL https://aclanthology.org/N16-1072.

[48] Avirup Sil, Gourab Kundu, Radu Florian, and Wael Hamza. Neural cross-lingual entity linking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. doi: 10.1609/aaai.v32i1.11964. URL https://ojs.aaai.org/index.php/AAAI/article/view/11964.

[49] Shyam Upadhyay, Nitish Gupta, and Dan Roth. Joint multilingual supervision for cross-lingual entity linking. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2495, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1270. URL https://aclanthology.org/D18-1270.

[50] Jan A. Botha, Zifei Shan, and Daniel Gillick. Entity Linking in 100 Languages. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7833–7845, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.630. URL https://aclanthology.org/2020.emnlp-main.630.

[51] Hamed Shahbazi, Xiaoli Z. Fern, Reza Ghaeini, Rasha Obeidat, and Prasad Tadepalli. Entity-aware elmo: Learning contextual entity representation for entity disambiguation, 2019.

[52] Ikuya Yamada, Koki Washio, Hiroyuki Shindo, and Yuji Matsumoto. Global entity disambiguation with bert, 2022.

[53] G P Shrivatsa Bhargav, Dinesh Khandelwal, Saswati Dana, Dinesh Garg, Pavan Kapanipathi, Salim Roukos, Alexander Gray, and L Venkata Subramaniam. Zero-shot entity linking with less data. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1681–1697, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.127. URL https://aclanthology.org/2022.findings-naacl.127.

[54] Eleni Partalidou, Despina Christou, and Grigorios Tsoumakas. Improving zero-shot entity retrieval through effective dense representations. In *Proceedings of the 12th Hellenic Conference on Artificial Intelligence*, SETN '22, New York, NY, USA, 2022. Association

for Computing Machinery. ISBN 9781450395977. doi: 10.1145/3549737.3549771. URL https://doi.org/10.1145/3549737.3549771.

[55] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, September 2019. ISSN 1367-4811. doi: 10.1093/bioinformatics/btz682. URL http://dx.doi.org/10.1093/bioinformatics/btz682.

[56] Emily Alsentzer, John R. Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B. A. McDermott. Publicly available clinical bert embeddings, 2019.

[57] Z. Ji, Q. Wei, and H. Xu. Bert-based ranking for biomedical entity normalization. *AMIA Jt Summits Transl Sci Proc*, pages 269–277, 2020.

[58] Shikhar Vashishth, Denis Newman-Griffis, Rishabh Joshi, Ritam Dutt, and Carolyn P. Rosé. Improving broad-coverage medical entity linking with semantic type prediction and large-scale datasets. *Journal of Biomedical Informatics*, 121:103880, September 2021. ISSN 1532-0464. doi: 10.1016/j.jbi.2021.103880. URL http://dx.doi.org/10.1016/j.jbi.2021.103880.

[59] Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. Self-alignment pretraining for biomedical entity representations. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4228–4238, June 2021.

[60] Elena Zotova, Aitor García-Pablos, Montse Cuadros, and German Rigau. Vicomtech at medprocner 2023: transformers-based sequence-labelling and cross-encoding for entity detection and normalisation in spanish clinical texts. In *CLEF 2023: Conference and Labs of the Evaluation Forum*, 2023.

[61] Afshin Rahimi, Timothy Baldwin, and Karin Verspoor. WikiUMLS: Aligning UMLS to Wikipedia via cross-lingual neural ranking. In Donia Scott, Nuria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5957–5962, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.523. URL https://aclanthology.org/2020.coling-main.523.

[62] Fangyu Liu, Ivan Vulić, Anna Korhonen, and Nigel Collier. Learning domain-specialised

representations for cross-lingual biomedical entity linking. In *Proceedings of ACL-IJCNLP 2021*, pages 565–574, August 2021.

[63] Mujeen Sung, Hwisang Jeon, Jinhyuk Lee, and Jaewoo Kang. Biomedical entity representations with synonym marginalization, 2020.

[64] Samy Ateia and Udo Kruschwitz. Is chatgpt a biomedical expert? – exploring the zero-shot performance of current gpt models in biomedical tasks, 2023.