Fall 2021

Enhance Representation Learning of Clinical Narrative with Neural Networks for Clinical Predictive Modeling

Yuqi Si

Enhancing Representation Learning of Clinical Narrative with Neural Networks for Clinical Predictive Modeling

By

Yuqi Si, MS

APPROVED:

Kirk Roberts, PhD, Chair

Xiaoqian Jiang, PhD

Digitally signed by ebernstam
DN: dc=edu, dc=uthouston, ou=People, ou=UTP, cn=ebernstam
Date: 2021.12.03 09:56:26 -06'00'

Elmer Bernstam, MD, MSE

Timothy Miller, PhD

Date approved: Dec 3, 2021

Enhancing Representation Learning of Clinical Narrative with Neural Networks for Clinical Predictive Modeling

A
Dissertation

Presented to the Faculty of
The University of Texas
Health Science Center at Houston
School of Biomedical Informatics
in Partial Fulfilment of the Requirements for the Degree of

Doctor of Philosophy

By

Yuqi Si, MS

University of Texas Health Science Center at Houston
2021

Dissertation Committee:

Kirk Roberts, PhD[1], Advisor
Xiaoqian Jiang, PhD[1]
Elmer Bernstam, MD, MSE[1,2]
Timothy Miller, PhD[3,4]

[1]School of Biomedical Informatics, UT Health
[2]Department of Internal Medicine, McGovern Medical School, UT Health
[3]Computational Health Informatics Program, Boston Children's Hospital
[4]Department of Pediatrics, Harvard Medical School

To my amazing parents, and my wonderful husband Kuo.

ACKNOWLEDGMENTS

Finishing a Ph.D. during a global pandemic is not easy. I am sincerely grateful to everyone who has supported and helped me throughout my PhD journey. First and foremost, this would not have happened without the support of my primary advisor, Dr. Kirk Roberts, who always granted me the freedom to pursue research topics based on my interests. He is one of the greatest clinical NLP experts, and I feel very honored to have worked with him over the past four years. His consideration, inspiration, intelligence, and support helped me grow substantially, both in research capabilities and in personality traits. He is not only a great scholar with immense knowledge, but also an incredibly amazing person. I can hardly think of a better PhD advisor than him.

Thank you to my committee members: Dr. Xiaoqian Jiang, Dr. Elmer Bernstam, and Dr. Timothy Miller. They have all provided insightful suggestions for my research. I want to thank Dr. Jiang for all the important discussions and providing his most powerful server when computational resources were highly demanded for large pre-trained models. So that I can work on almost any experiment I want without too much worry about resources. Thank you to Dr. Bernstam, who is the only medical expert on the team and let me realize there were a lot of realistic problems with some of the things that I was doing, and has proven very beneficial to my work. Thanks to my external committee member, Dr. Miller, I was always inspired a lot by reading his research publications. Also, with his valuable feedback, I could develop new approaches to address the problems.

At various phases of my study journey over the previous years, I have gained training and guidance from numerous faculty and researchers in the clinical informatics field, from UT Health and Columbia University. In particular, I would like to acknowledge Dr. Chunhua Weng and Dr. Nicholas Tatonetti, who initially led

me to the path of informatics research. Also, I feel honored to work and study with these amazing mentors – Dr. Susan Fenton, Dr. Hua Xu, Dr. Trevor Cohen, and Dr. Ning Shang. Thanks for their consideration and commitment to improving my studies and research.

Special appreciation to my colleagues and coauthors at SBMI for sharing their knowledge and time – Surabhi Datta, Sarvesh Soni, Dr. Jingcheng Du, Jingqi Wang, and all the innovative minds that are part of the SBMI family. Also, to all the academic affairs staff and administrators at SBMI – Jaime Hargrave, Chelsea Overstreet, Miranda Hedrick, and Susan Rojas. I really appreciate all of their prompt help to make my life a lot easier.

My PhD studies would not even be possible without numerous sources of financial support. I truly appreciate the funding agencies, including the National Institutes of Health (NIH), the National Library of Medicine (NLM), the Cancer Prevention and Research Institute of Texas (CPRIT), and the Patient-Centered Outcomes Research Institute (PCORI), as well as the scholarship programs, including the SBMI Doctoral Fellowship, Willerson Endowed Scholarship, James A. Baker Transformation and Hope Scholarship, Mercedes-Benz Star Motor Cars Scholarship, and Dr. Aoki Travel Award. Last but not least, I am deeply in debt to my family for their unconditional love and support. I am really blessed to have such a wonderful family, and they are my greatest source of inspiration for pursuing my goals. My parents' unwavering love and strict discipline is the motivation that has taught me to work hard for the things that I want to achieve. My husband's continuous devotion and encouragement has helped me get through many challenging times. Without him, I would not have finished this PhD journey. I find it hard to express my gratitude because it is so boundless, but I am truly thankful to have my family in my life.

ABSTRACT

Medicine is undergoing a technological revolution. Understanding human health from clinical data has major challenges from technical and practical perspectives, thus prompting methods that understand large, complex, and noisy data. These methods are particularly necessary for natural language data from clinical narratives/notes, which contain some of the richest information on a patient. Meanwhile, deep neural networks have achieved superior performance in a wide variety of natural language processing (NLP) tasks because of their capacity to encode meaningful but abstract representations and learn the entire task end-to-end. In this thesis, I investigate representation learning of clinical narratives with deep neural networks through a number of tasks ranging from clinical concept extraction, clinical note modeling, and patient-level language representation. I present methods utilizing representation learning with neural networks to support understanding of clinical text documents.

I first introduce the notion of representation learning from natural language processing and patient data modeling. Then, I investigate word-level representation learning to improve clinical concept extraction from clinical notes. I present two works on learning word representations and evaluate them to extract important concepts from clinical notes. The first study focuses on cancer-related information, and the second study evaluates shared-task data. The aims of these two studies are to automatically extract important entities from clinical notes. Next, I present a series of deep neural networks to encode hierarchical, longitudinal, and contex-

tual information for modeling a series of clinical notes. I also evaluate the models by predicting clinical outcomes of interest, including mortality, length of stay, and phenotype predictions. Finally, I propose a novel representation learning architecture to develop a generalized and transferable language representation at the patient level. I also identify pre-training tasks appropriate for constructing a generalizable language representation. The main focus is to improve predictive performance of phenotypes with limited data, a challenging task due to a lack of data.

Overall, this dissertation addresses issues in natural language processing for medicine, including clinical text classification and modeling. These studies show major barriers to understanding large-scale clinical notes. It is believed that developing deep representation learning methods for distilling enormous amounts of heterogeneous data into patient-level language representations will improve evidence-based clinical understanding. The approach to solving these issues by learning representations could be used across clinical applications despite noisy data. I conclude that considering different linguistic components in natural language and sequential information between clinical events is important. Such results have implications beyond the immediate context of predictions and further suggest future directions for clinical machine learning research to improve clinical outcomes. This could be a starting point for future phenotyping methods based on natural language processing that construct patient-level language representations to improve clinical predictions. While significant progress has been made, many open questions remain, so I will highlight a few works to demonstrate promising directions.

VITA

Ph.D. in Biomedical Informatics Sep. 2017 – Present

University of Texas Health Science Center

M.S. in Biomedical Engineering Sep. 2015 – Feb. 2017

Columbia University

B.E. in Biomedical Engineering Sep. 2013 – Jun. 2015

Sun Yat-sen University.

B.S. in Biology Sep. 2010 – Jun. 2014

Sun Yat-sen University.

FIRST-AUTHOR PUBLICATIONS

[1] **Si Y**, Bernstam E, Roberts K. Generalized and Transferable Clinical Language Representation for Phenotyping with Limited Data. *Journal of Biomedical Informatics.* 2021 Apr 1;116:103726.

[2] **Si Y**, Du J, Li Z, Jiang X, Miller T, Wang F, Zheng WJ, Roberts K. Deep Representation Learning of Patient Data from Electronic Health Records (EHR): A Systematic Review. *Journal of Biomedical Informatics.* 2021 Mar 1; 115: 103671.

[3] **Si Y**, Roberts K. Hierarchical Transformer Networks for Longitudinal Clinical Document Classification. *arXiv preprint* arXiv:2104.08444. 2021.

[4] **Si Y**, Roberts K. Patient Representation Transfer Learning from Clinical Notes based on Hierarchical Attention Mechanism. *AMIA Joint Summits on Translational Science proceedings.* 2020:597–606.

[5] **Si Y**, Wang J, Xu H, Roberts K. Enhancing clinical concept extraction with contextual embeddings. *Journal of the American Medical Informatics Association*. 2019 Jul 2; 26(11):1297-304.

[6] **Si Y**, Roberts K. Deep Patient Representation of Clinical Notes via Multi-Task Learning for Mortality Prediction. *AMIA Joint Summits on Translational Science proceedings. 2019 May 6;2019:779-88.*

[7] **Si Y**, Roberts K. A Frame-Based NLP System for Cancer-Related Information Extraction. *InAMIA Annual Symposium proceedings.* AMIA Symposium 2018 (Vol. 2018, pp. 1524-1533).

[8] **Si Y**, Weng C. An OMOP CDM-based Relational Database of Clinical Research Eligibility Criteria. *Studies in health technology and informatics.* 2017;245:950-4.

CO-AUTHOR PUBLICATIONS

[1] Du J, Xiang Y, Sankaranarayanapillai M, Zhang M, Wang J, **Si Y**, et al. Extracting Post-marketing Adverse Events from Safety Reports in the Vaccine Adverse Event Reporting System (VAERS) using Deep Learning. *Journal of the American Medical Informatics Association*. 2021 Feb 27.

[2] Champagne-Langabeer T, Swank M, Manas S, **Si Y**, Roberts K. Dramatic Increases in Telehealth-Related Tweets During the Early COVID-19 Pandemic: A Sentiment Analysis. Accepted and forthcoming in Healthcare. 2021.

[3] Greenspan N, **Si Y**, Roberts K. Extracting Concepts for Precision Oncology from the Biomedical Literature. *AMIA Joint Summits on Translational Science proceedings.* 2021.

[4] Miller T, Dligach D, Wang F, **Si Y**, Meric-Bernstam F. Learning Patient Representations from Electronic Health Records across Diverse Data Types. AMIA. 2020.

[5] Wu S, Roberts K, Datta S, Du J, Ji Z, **Si Y**, Soni S, Wang Q, Wei Q, Xiang Y, Zhao B. Deep learning in clinical natural language processing: a methodical review. *Journal of the American Medical Informatics Association*. 2020 Mar 1;27(3):457-70.

[6] Datta S, **Si Y**, Rodriguez L, Shooshan SE, Demner-Fushman D, Roberts K. Understanding spatial language in radiology: representation framework, annotation, and spatial relation extraction from chest X-ray reports using deep learning[J]. Journal of biomedical informatics, 2020, 108: 103473.

[7] Wei Q, Ji Z, **Si Y**, Du J, Wang J, Tiryaki F, Wu S, Tao C, Roberts K, Xu H. Relation Extraction from Clinical Narratives Using Pre-trained Language Models. *InAMIA Annual Symposium proceedings.* AMIA Symposium 2019.

[8] Xiang Y, Xu J, **Si Y**, Li Z, Rasmy L, Zhou Y, Tiryaki F, Li F, Zhang Y, Wu Y,

Zheng W, Zhi D, Tao C, Xu H and Jiang X. Time-sensitive Clinical Concept Embeddings Learned from Large Electronic Health Records. *BMC medical informatics and decision making.* 2019 Apr; 19(2): 58.

[9] Roberts K, **Si Y**, Gandhi A, Bernstam E. A FrameNet for Cancer Information in Clinical Narratives: Schema and Annotation. *InProceedings of the Eleventh International Conference on Language Resources and Evaluation.* 2018 May

[10] Butler A, Wei W, Yuan C, Kang T, **Si Y**, Weng C. The Data Gap in the EHR for Clinical Research Eligibility Screening. *AMIA Joint Summits on Translational Science proceedings.* 2018 May 18;2017:320-9.

[11] Han B, Long W, He J, Liu Y, **Si Y**, Tian L. Effects of dietary Bacillus licheniformis on growth performance, immunological parameters, intestinal morphology and resistance of juvenile Nile tilapia (Oreochromis niloticus) to challenge infections. *Fish&shellfish immunology.* 2015 Oct 1;46(2):225-31.

Field of Study

Biomedical Informatics

TABLE OF CONTENTS

Page

LIST OF TABLES . . . . . **xv**

LIST OF FIGURES . . . . . **xvi**

1 INTRODUCTION **1**

1.1 Background 1
1.2 Problem Definition 3
1.3 Motivations 5
1.4 Thesis Outline 6

2 LITERATURE REVIEW **9**

2.1 Representation Learning 9
2.2 Representation Learning in Natural Language Processing 10
2.2.1 Count-based Word Representation 13
2.2.2 Prediction-based Word Representation 15
2.2.3 Language model-based Representation 17
2.3 Representation Learning of EHRs Data 19
2.3.1 Temporal Matrix-based Patient Representation 20
2.3.2 Vector-based Patient Representation 21
2.3.3 Tensor-based Patient Representation 22
2.3.4 Graph-based Patient Representation 23
2.3.5 Sequence-based Patient Representation 24

3 CLINICAL WORD REPRESENTATION **27**

3.1 Cancer-related Information Extraction 28
3.1.1 Introduction 28
3.1.2 Related Work 30

3.1.2.1 Cancer Information Extraction 30
3.1.2.2 Deep Learning Models for Biomedical Texts . . . 32
3.1.3 Methods 34
3.1.3.1 Dataset 34
3.1.3.2 Model 36
3.1.3.3 Experiments 39
3.1.4 Results 41
3.1.5 Discussion 44
3.2 Concept Extraction with Contextual Embeddings 46
3.2.1 Introduction 46
3.2.2 Background 47
3.2.2.1 Word Embeddings 47
3.2.2.2 Language Model-based Embeddings 48
3.2.2.3 Clinical Concept Extraction 49
3.2.3 Methods 50
3.2.3.1 Embeddings 50
3.2.3.2 Datasets 52
3.2.3.3 Experiment Details 53
3.2.4 Results 55
3.2.4.1 Performance Comparison 55
3.2.4.2 Pretraining Evaluation 55
3.2.5 Discussion 57

4 CLINICAL NOTE REPRESENTATION 59

4.1 Hierarchical Convolutional Neural Network 59
4.1.1 Motivation 59
4.1.2 Model Architecture 60
4.2 Hierarchical Attention Network 61
4.2.1 Motivation 61
4.2.2 Model Architecture 62
4.2.3 Implementation Details 64
4.2.4 Performance Comparison 65
4.3 Hierarchical Transformer Network 66

4.3.1 Motivation . . . 66
4.3.2 Model Architecture . . . 68
4.3.2.1 Word-level BERT . . . 68
4.3.2.2 Sentence- and Document-level Transformers . . . 70
4.3.2.3 Adaptive Segmentation and Filling . . . 70
4.3.3 Experiment Details . . . 72
4.3.3.1 Dataset and Prediction Tasks . . . 72
4.3.3.2 Implementation details . . . 74
4.3.4 Performance Comparison . . . 75
4.3.4.1 Compared Baselines . . . 75
4.3.4.2 Evaluation Metrics . . . 76
4.3.4.3 Results . . . 77
4.3.5 Ablation Study . . . 79
4.3.5.1 Input Text Lengths . . . 79
4.3.5.2 BERT Variations . . . 81
4.3.5.3 Transformer Encoder Variations . . . 84

5 PATIENT-ORIENTED REPRESENTATION . . . 87

5.1 Multi-task Learning . . . 89
5.1.1 Backgrounds . . . 89
5.1.2 Related Work . . . 90
5.2 Transfer Learning . . . 92
5.2.1 Backgrounds . . . 92
5.2.2 Related Work . . . 93
5.3 Generalized and Transferable Patient Language Representation for Phenotyping with Limited Data . . . 95
5.3.1 Introduction . . . 95
5.3.2 Methods . . . 98
5.3.2.1 Overall Framework . . . 98
5.3.2.2 High-prevalence Phenotyping-guided Pre-training 100
5.3.2.3 Fine-tuning on Low-prevalence Conditions . . . 101
5.3.3 Experiments . . . 106
5.3.3.1 Baseline Methods . . . 106

5.3.3.2 Data and Implementation Details 107
5.3.3.3 Evaluation Metrics 107
5.3.4 Results 108
5.3.4.1 Pre-training Results 108
5.3.4.2 The Effectiveness of Pre-training 110
5.3.4.3 The Effectiveness of MTL 114
5.3.5 Discussion 117

6 CONCLUSION 120

6.1 Thesis Overview 120
6.2 Significance and Contribution 123
6.3 Limitations and Future Directions 124

References 128

LIST OF TABLES

Table		Page
3.1	Frame Lexical Units and Elements	35
3.2	Descriptive Statistics of Concepts	40
3.3	Performance of System for Frame Identification	42
3.4	General Performance Evaluation of System for Element Classifier	42
3.5	Cross-Frame Per Category Performance Evaluation of System for Element Classifier	43
3.6	Descriptive Statistics of Concepts	53
3.7	Resources of Off-the-shelf Embeddings	54
3.8	Test Set F1 Comparisons in Exact Matching	56
4.1	Performance of Different Models on Mortality Prediction Tasks	66
4.2	Descriptive Statistics of Datasets	73
4.3	Descriptive Statistics of Phenotypes	74
4.4	Overall Performance Comparisons	77
4.5	PRC scores of Different Models for All Phenotypes	79
4.6	Performance of Hypertension with Different Input Lengths	80
4.7	Performance of Hypertension with Distilled BERT Models	82
4.8	Performance of Hypertension with Transformer Variations	85
5.1	Top Five High-prevalence Phenotypes in Three Organ Systems	100
5.2	Low-prevalence Phenotypes in the Circulatory System	103
5.3	Low-prevalence Phenotypes in the Respiratory System	104
5.4	Low-prevalence Phenotypes in the Genitourinary System	105
5.5	Performances of High-prevalence Phenotypes	109
5.6	Predictive Performances of the Circulatory System	111
5.7	Predictive Performances of the Respiratory System	112
5.8	Predictive Performances of the Genitourinary System	113
5.9	Comparisons of MTL with Baselines	114
5.10	Number of Phenotypes for Best Performances and Tolerable Cases	115
5.11	Average Mean Squared Error across Organ Systems	116

LIST OF FIGURES

Figure Page

1.1 Natural Language Processing Stacks 4
1.2 Complex Medical Word Embeddings 5

2.1 Pipeline of Patient Representation Learning 20

3.1 Main Architecture of the System. 38
3.2 Performances on the i2b2 2010 Task by Pre-training Steps 57

4.1 Hierarchical Convolutional Neural Network Model Architecture . 60
4.2 Hierarchical Attention Network Model Architecture 63
4.3 Hierarchical Transformer Network Model Architecture 69

5.1 Clinical Outcome-targeted Supervised Learning 88
5.2 Multi-task Learning Joint Optimization 89
5.3 Transfer Learning in NLP 93
5.4 Overall Pre-training and Fine-tuning Architecture 98
5.5 Box-plots of AUC Distributions 110

CHAPTER 1

INTRODUCTION

1.1 Background

The field of Natural Language Processing (NLP) is continually growing to encompass new formats of text, new types of information, and new domain applications. When one develops NLP approaches, it enables a computer to process and analyze large amounts of natural language data. Teaching computers to comprehend materials written in human language is one of the most challenging and fundamental problems for NLP. Also, NLP is constantly applied in the fields of biomedical and healthcare, with diverse and innovative applications to biomedical research and clinical practice.

Understanding the characteristics of medical language and incorporating these patterns into model development is critical for biomedical language processing. Notably, the enhancement of representation learning is an important method development that aids in the achievement of these factors. Essentially, representation learning is an important approach for encoding "hidden" knowledge–that is, knowledge readily available to human experts but not directly stated in the text. It aims to learn meaningful information from raw data automatically. It also determines how much useful information can be extracted from raw data for further classification or prediction. Furthermore, representation learning makes sense because it is coherent with the patterns hidden in the data, where each sample can be represented

by patterns across many computational elements, where each element is engaged in representing different samples.

At present, representation learning is primarily performed using advanced deep learning methods. In order to understand why deep learning is needed in representation learning, I introduce the basic idea of deep learning: deep learning models make predictions through an iterative training process, during which the input data is repeatedly fed into the model, hyper-parameters are gradually changed, and eventually they learn to connect the input data into good predictions. Each layer, in effect, learns to make the next layer's prediction a little easier, so each layer contains meaningful information with dense vectors.

Deep learning has gradually emerged as a fundamental method for mining information from large amounts of data in the era of artificial intelligence (AI). Specifically, deep learning has facilitated the development of key applications such as computer vision, language understanding, and speech recognition. In contrast to typical machine learning methods, deep learning is a paradigm for automatic training across large datasets. Deep learning also takes full advantage of massively-growing computational resources and web-scale data collection. State-of-the-art deep learning models are being increasingly developed for unstructured data such as text. For instance, convolutional neural networks (CNNs) built for matrix-format data and recurrent neural networks (RNNs) built for sequential data are both utilized to model natural language.

In terms of the reason why deep learning is necessary for representation learning, these models have deep architectures that are able to capture abstract and dense

features from raw data. Also, deep learning models are built with a large number of non-linear calculations, which would generate more fine-grained features through these complex non-linear relations. And eventually, deep learning can build more effective representations that connect input towards effective predictions.

1.2 Problem Definition

Utilizing electronic health records (EHRs) for clinical outcome tasks has been widely studied, ranging from predicting patient health conditions, such as disease and mortality risk predictions, to monitoring patient trajectories such as readmission and length-of-stay forecasting. Even though structured EHRs contain sufficient information about patient encounters, free-text clinical notes provide specific details that are more fine-grained to understand patient trajectories. These notes typically provide information that is supplementary to other data sources, and the information has enormous promise if appropriately employed. Existing methods, on the other hand, either only applied methods for extracting concepts and using these concepts as features, or solely implemented topic modeling to comprehend the subjects in clinical notes.

There are still many opportunities for advancement in terms of establishing a more comprehensive knowledge of medical language and building well-suited models for a holistic patient-level view of clinical notes.

Furthermore, clinical NLP also attempts to build computational algorithms for machines to understand clinical documents. Those documents are unstructured free-text with different granularities and complex vocabularies for multiple tasks.

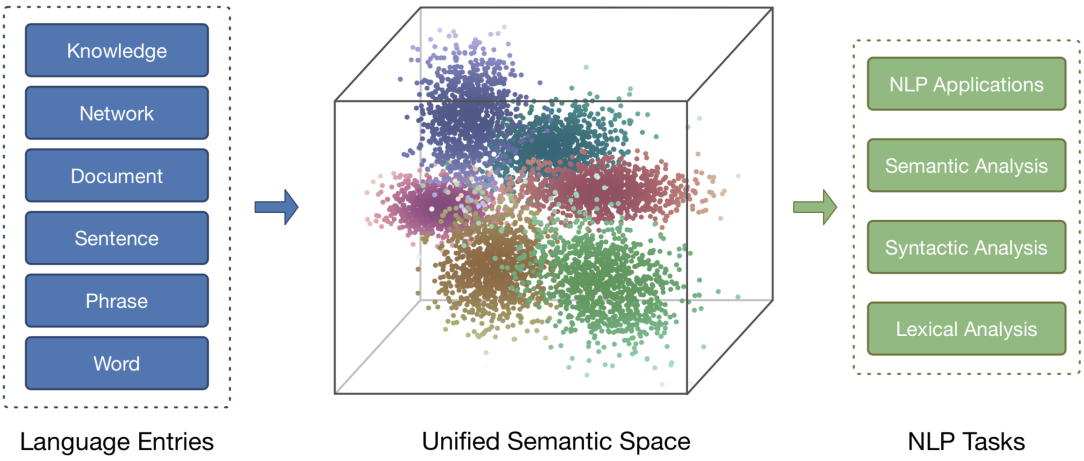

Figure 1.1: Natural Language Processing Stacks

With regards to how deep representation learning assists with clinical NLP, essentially, clinical NLP concerns multiple levels of language stacks, including but not limited to characters, words, phrases, sentences, paragraphs, sections, and documents. Deep representation learning can help to represent the semantics of these language stacks in unified semantic spaces and, at the same time, build complex semantic relations among these language stacks. Also, as discussed earlier, medical text is characterized by its complexity in vocabulary and richness in morphology. Different approaches to representing learning would help to facilitate understanding across this complex and rich contextual information.

Specifically, the processing of clinical documents has special difficulties. For instance, usage of grammatically incorrect language, abbreviations, and terms that are domain-specific. Methods for modeling clinical notes should have sufficient exposure to basic **domain knowledge**, which can be either direct or indirect association with the method development. Besides, another major downside in the

Figure 1.1 retrieved from Liu et al. (2020).

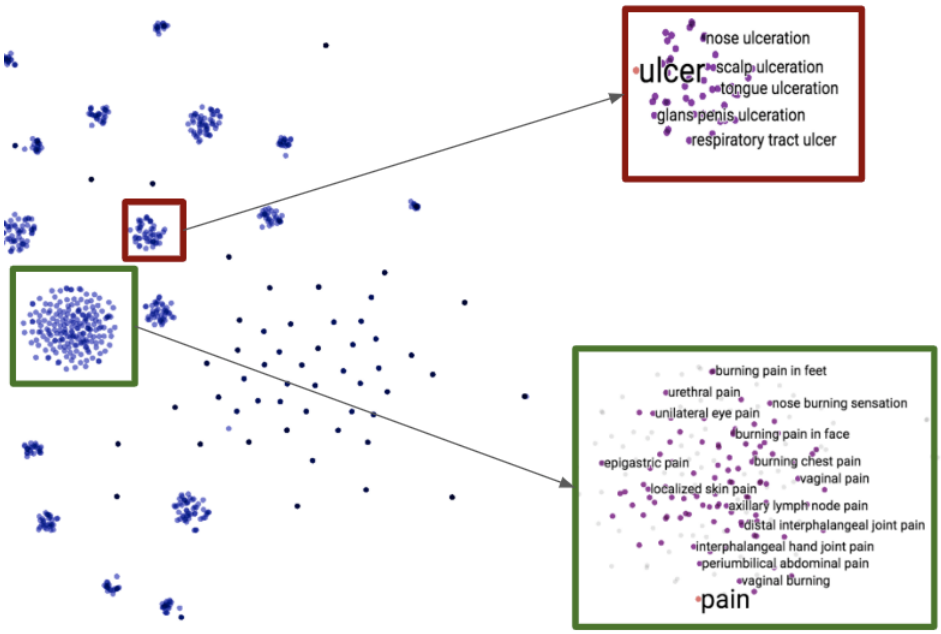

Figure 1.2: Complex Medical Word Embeddings

development of cutting-edge deep learning models for clinical NLP tasks is that the method fails to incorporate **essential characteristics** of clinical notes into the model capabilities. In the end, before implementing models for decision making in clinical practice, it is critical to make sure that the models achieve a high level of performance while also being **generalized** and **transferable** across varied clinical tasks.

1.3 Motivations

In this thesis, I particularly investigate NLP for clinical notes in the EHR. I focus my efforts on applying state-of-the-art deep representation learning methods

Figure 1.2 retrieved from Amatriain (2020).

in the NLP domain to clinical NLP tasks in order to improve medical language understanding beyond its current state. One can build deep representation learning to model clinical notes for a wide variety of clinical outcome predictions, and also proceed towards enabling representations to simulate the actual characteristics of clinical notes. Additionally, I investigate the adoption of external free-text resources that include medical knowledge to enrich medical information hidden in the models for clinical language understanding.

Lastly, I develop models that learn semantic representations from patient notes that cover as much information as possible from a holistic aspect. I focus on representation learning approaches with transfer learning to make full use of clinical notes by mapping notes towards clinical outcomes directly. My ultimate goal is to develop meaningful representations that can be transferred to multiple clinical tasks. Such representations not only include clinically relevant information about specific patients, but they also associate similar patients with similar patterns to a shared latent space.

1.4 Thesis Outline

CHAPTER 2 reviews the background of this thesis. This chapter covers some of the conceptual knowledge of representation learning and its application in natural language processing and EHR data modeling, particularly focusing on methods using advanced deep learning models. From a methodological standpoint, I systematically review this topic and include both qualitative and quantitative assessments. More importantly, I demonstrate the necessity and feasibility of learning represen-

tations from natural language and patient data.

CHAPTER 3 discusses the development of clinical word representations for clinical concept extraction. This includes work that proposes a frame-based NLP system to identify cancer-related information from clinical notes. I implement a bidirectional long short-term memory with conditional random field (Bi-LSTM-CRF) and incorporate both character and word representations into the model input. This study shows the usefulness of different representations combined with deep learning models for extracting frame semantic information from clinical notes. Another study in this chapter explores possible improvements in using contextual representations for clinical concept extraction by comparing these methods to traditional word representation methods. I investigate best practices for implementing these recent state-of-the-art contextual representations into clinical tasks. The best-performing representations outperform existing state-of-the-art methods that achieve clinical concept extraction tasks.

CHAPTER 4 presents a series of hierarchical neural networks for modeling clinical notes over a long time scale. To explicitly learn representations from long-sequence clinical notes, I develop models initialized by CNNs, then Hierarchical Attention Networks (HANs), and finally Hierarchical Transformer Networks. For CNNs, I also implement target replication that incorporates the final loss with the loss at intermediate steps, so as to emphasize relations between sentences. For HANs, I further apply an adaptive segmentation module to differentiate short-period co-occurrences with long-term dependencies in clinical note sequences. To this end, I consider three aspects of clinical notes – contextual, hierarchical, and

longitudinal – and propose Hierarchical Transformer Networks. I evaluate these models on real clinical outcome predictions containing mortality and phenotype predictions.

CHAPTER 5 embarks on a novel representation learning architecture, with the multi-task pre-training and transfer learning, to learn generalized and transferable patient representations. The model is pre-trained with a range of high-prevalence phenotypes before being fine-tuned towards downstream tasks. I validate the impact this representation can have on low-prevalence phenotypes, as it is a challenging task because of the limited data. The results lead us to conclude that this multi-task supervised pre-training is a solid and robust method for learning generalized patient representations for numerous phenotypes.

CHAPTER 6 concludes the thesis. I also discuss the limitations and propose a few directions for future study.

CHAPTER 2

LITERATURE REVIEW

2.1 Representation Learning

Representation learning is established as learning representations of raw data automatically to extract meaningful information for building effective classifiers or predictors (Bengio et al., 2013). Each sample can be represented by a pattern across numerous computational elements, and each element is engaged in representing different samples. Discovering spaces where data can be separated linearly, differentiating invariant attributes of objects from noisy properties, and categorizing high-dimensional data into interpretable clusters are all representation learning problems. A typical machine learning algorithm requires three main components: representation, model, and objective. That means we need to first convert raw data into meaningful representation, then choose an optimal model, and finally train the model to meet the requirements of the objective function. The performance of machine learning algorithms largely depends on the effectiveness of data representation whether we can disentangle the underlying factors hidden in the data.

Effective representations convey meaningful knowledge, which means that a trained representation of a compact size is able to capture essential input patterns. The motivation of representation learning is to make patterns underlain by data clear. Traditional machine learning requires careful feature engineering in preprocessing. This makes a certain task need customized and hand-crafted features,

but the efforts in this step are time-consuming and inflexible. Deep learning, on the other hand, is constructed with the composition of non-linear transformations and would yield more abstract but fine-grained, and eventually more effective representations. While specific domain knowledge can be utilized in developing representations, learning with generic underlying patterns would be much more generalized and applicable for numerous tasks, and the application of deep learning is motivating the development of more effective representation learning methods for a wide variety of work.

2.2 Representation Learning in Natural Language Processing

Natural language texts are typically unstructured data with varying granularities of information in multiple domains. The goal of representation learning for natural language is to automatically represent texts on numerical scales. Modern NLP is mainly based on representation learning. Natural language representation focuses on learning generic or static representations with the motivation to improve the end task, which significantly minimizes the need for humans to curate manual features. The ability to efficiently consume large amounts of data to learn generic features is important to better suit the downstream NLP tasks. Another major reason for representation learning in NLP fields is that textual data is fundamentally different from structured data such as images (i.e., the fixed size of two-dimensional vectors), where text has to be learned sequentially to be captured with variable sequence lengths. This has inspired the wide adoption of current trends in the NLP community to embrace representation learning approaches. The investigation into

representation learning in NLP has progressed significantly since then (Liu et al., 2020).

In conventional NLP disciplines, classic ideas and approaches, such as n-gram and bag-of-word, have been previously applied in numerous applications. However, these methods always suffer from the notorious *curse of dimensionality* in large-scale corpora. They are considered to have relatively limited capabilities when it comes to analyzing larger text objects (i.e., documents) and extracting semantically-meaningful information. Such methods are also constrained by the fact that they can only analyze information that they already perceive. Advanced computational models seek to minimize this gap by simulating how humans understand language, for instance, by modeling semantic features that are implicitly expressed in the language.

An alternative approach to such representation is known as distributed representation (Hinton et al., 1986), where concepts are represented as low-dimensional and real-valued vectors. Many early approaches of word representations are built on the assumption of this distributional representation. Accordingly, two words that appear frequently in analogous linguistic environments are more semantically related, so both of them should be closer to each other in the embedding space. The word representations generate low-dimensional dense vectors that substantially improve the curse of dimension and sparsity of the traditional bag-of-word method. The distributed word representations are particularly good at capturing subtle semantic similarities, and the downstream tasks relying on those representations have produced state-of-the-art results.

While these distributed word representations are important in many clinical NLP tasks, they are still limited in their capability to represent words with multiple meanings in varied contexts. They encompass all probable definitions of one word in a single embedding, and the embedding is not context-aware. They also face a major out-of-vocabulary issue. One milestone for this issue is the introduction of pre-trained language models. Recent research in computer vision has established the use of transfer learning, in which huge CNN models are pre-trained on large-scale image recognition datasets annotated by humans, the ImageNet. Through leveraging ImageNet's extensive visual data, fine-tuning these pre-trained models with a small amount of task-specific data enables the models to perform effectively on downstream tasks. This initiates the first stage of investigations into pre-trained language models in the NLP domain. The Transformers were further proposed to enable enormously deep neural models for NLP tasks. When the size of the pre-trained language models is increased, such large-scale models with millions of hyper-parameters are able to derive information including word disambiguation, semantic and syntactic structures, and underlying information from the context. Existing large-scale language models have prompted model performance for a wide range of NLP tasks due to their generalizability and robustness. By extending this process, we may potentially gain a better insight into how human language processing works.

2.2.1 Count-based Word Representation

Conventional NLP algorithms rely heavily on count-based word representation learning. The word frequency and co-occurrence matrix are widely used in count-based methods, assuming that words in similar contexts have similar count-based statistics. The count-based method projects those statistics into feature vectors of individual words.

The simplest way to automatically represent a word using a numerical feature is the bag-of-words (BoW) (Zhang et al., 2010). Each word is represented by a $1 \times$ N matrix, where N is the vocabulary size. The position corresponding to an individual word is assigned a value of 1, while all other positions in this vector are assigned a value of 0, which we also identify as one-hot encoding. This method is a simplifying method used in NLP and information retrieval (IR). However, it fails to consider grammar related to word sequence, or any semantic meaning of words, and just distinguishes words from each other. As there are millions of words in the corpus, the vectors tend to be extremely sparse, resulting in the curse of dimension. In addition, the methods are limited in their ability to generalize to out-of-vocabulary words and easily overfit to the training corpus (Wallach, 2006).

N-gram models are one of the earliest ideas related to count-based word representation learning (Cavnar et al., 1994). N-grams are consecutive sequences of one or more tokens derived from texts. The n-gram models can be used as probabilistic language models to predict the next tokens in a sentence within a Markov chain. Intuitively, we normally refer to certain previous words to predict the next word in a sentence (i.e., previous n-1 tokens in the case of an n-gram model). And if we pro-

cess a large-scale corpus, we may calculate and obtain a probability estimation for each token, assuming all possible combinations of the n-1 previous tokens that have been encountered. These probabilistic language models are important for predicting words in sequences and also for developing vector representations of words, as they may encode the meanings of words in their respective vector representations.

TF-IDF, short for term frequency–inverse document frequency, is a numerical measure of a word's importance in a textual collection (Ramos et al., 2003). It is widely applied as a weighting method in IR and NLP. In real-world scenarios, some words are quite frequently used (e.g., "the", "is", "and", etc). However, they contain very little information about the actual knowledge of the document. Raw frequency is not the best indicator of word association. The TF-IDF weighting is a numerical way to evaluate the importance of a word to a document in large-scale textual collections. The importance of a word increases proportionally to the frequency of a word in the entire document set, but decreases by the frequency of a word in a single document. Formally, this is achieved by multiplying term frequency and inverse document frequency. The former is defined as word frequency in a single document. The latter is measured by dividing the total number of documents by the number of documents actually containing that word, and then calculating the logarithm. The TF-IDF weighting is another way to calculate co-occurrence statistics, and it plays an important role in many aspects of IR and NLP. It is considered as a baseline and straightforward method to try at the beginning.

2.2.2 Prediction-based Word Representation

The prior methods learn how to represent words as sparse and long vectors with dimensions related to vocabulary size or the number of documents. A more effective and powerful form of word representation learning, word embeddings, is mostly derived from prediction models to generate short and dense vectors. These embeddings have dimensions ranging from 50 to 1000, as opposed to long and sparse vectors. Besides, unlike the vectors that are mostly zeros in count-based word representations, prediction-based word vectors consist of continuous numerical values. Over time, word embeddings have performed well in a wide range of NLP tasks. Because these dense vectors perform better at capturing words with similar meanings, while the count-based representations fail to distinguish between them. There are two typical algorithms for computing such word embeddings: Skip-gram and Continuous Bag of Words (CBOW). Both of them implemented self-supervision to learn from their own data without any need for manual effort in labeling. They are also known as word2vec (Mikolov et al., 2013b). The intuitions behind the algorithms are shallow neural networks based on distributional hypothesis. CBOW learns the embeddings in such a way that they can predict the target word from words in the context (Mikolov et al., 2013b). Skip-gram generates the embeddings that can be used to predict the context given the target word. There are also advanced variants of these two algorithms to incorporate more information into the model. Take the basic Skip-gram as an example to illustrate training steps more specifically: The training system first considers the target word and its context words (with a context window) as positive instances, and then randomly selects

sample words from the lexicon to obtain negative instances. Then the classifier will be trained iteratively to distinguish between these two cases, and the final trained weights will be used as the word embeddings.

Another widely known static embedding approach is GloVe, an abbreviation for Global Vectors for Word Representation (Pennington et al., 2014). GloVe is proposed to capture global corpus statistics. Combining the intuitions of count-based methods with the word co-occurrence matrix, it also captures the hidden structures that are implemented by word2vec. The GloVe is built on word-context matrices with matrix factorization. It constructs a large co-occurrence matrix (words $\times$ contexts). Specifically, for each word (i.e., row), the entry for each word is the number of times this word occurs in the corresponding context (i.e., column) in large corpora. The column size is quite large as it is essentially combinatorial. As a result, we must factorize this large matrix to produce a low-dimensional matrix with the size of word $\times$ features. Each row now represents a dense vector for the corresponding word. The matrix factorization is achieved by minimizing the reconstruction loss, similar to a dimension reduction algorithm, where the loss attempts to find the representations that could encode the most variance in the original data. Overall, GloVe is a probabilistic model based on ratios of probabilities from the co-occurrence matrix.

Despite the differences in methodological aspects, all of these approaches are relatively efficient to train, make extensive use of large-scale corpora, and have been effectively applied to numerous NLP tasks. As the key component of the NLP pipeline, word representations transform discrete words into low-dimensional

dense vectors with encoded information. In general, word representation learning is considered a fundamental step and enables the computer to better compute and understand natural language than prior to it.

Nevertheless, compared to the neural network language models that have established the state-of-the-art in recent studies, the word embedding methods are much simpler and shallower in two aspects. It simplifies the task, making the task a binary classification rather than a sequential word prediction. Such word embeddings are static embeddings where the vector for each word is fixed. Also, the network architectures are shallow networks rather than RNNs or even Transformers that require more complex training mechanisms.

2.2.3 Language model-based Representation

Many recent NLP works have been inspired by the potential of transfer learning, in which large-scale models are first pre-trained to leverage rich knowledge and then fine-tuned and adjusted to the downstream target task.This approach has achieved superior performance on many previous tasks and fueled an innovative paradigm shift in the NLP field. Language modeling is not a new idea. (Collobert and Weston, 2008) originally proposed to pre-train a model on a variety of NLP tasks to learn features rather than hand-crafting them, as the latter was the common method at that time. They employed language modeling as one of many supervised tasks in a multi-task learning scenario, along with part-of-speech tagging, named entity recognition, and semantic role labeling.

Language modeling did not gain widespread application in NLP until the advent

of the transfer learning recipe of pre-training and fine-tuning. In pre-training, the algorithm is trained to optimize and mimic the nuances of language, so as to build a general-purpose language model that can describe what natural language looks like. The training algorithm generally includes objective functions such as masked-language modeling and next sentence prediction. Such state-of-the-art pre-trained models in NLP are established through representation transfer and model transfer. For example, ELMo was a 2-layer Bi-LSTM language model (bi-LM) and the contexual embeddings are extracted with trainable combinations of the inner state state of the bi-LM, which is a type of representation transfer (Peters et al., 2018). In contrast to ELMo, BERT employs model transfer, which dynamically changes the parameters of the entire language model on the target task Devlin et al. (2019). As a result, the pre-trained BERT model can be fine-tuned by adding only one additional layer to achieve state-of-the-art performances for a variety of tasks. Inspired by BERT, many more effective pre-trained language models for NLP tasks have been proposed lately. In addition to BERT, there is a vast family of models that have evolved from it, including RoBERTa (Liu et al., 2019), XLNet (Yang et al., 2019), Transfomer-XL (Dai et al., 2019), SpanBERT (Joshi et al., 2020), ELECTRA (Clark et al., 2020), ALBERT (Lan et al., 2019), ERNIE (Zhang et al., 2019), etc. Researchers now exploit existing language models for numerous NLP tasks, prompting them to accomplish the desired task or reformulating the task as a text generation problem and applying language models to solve it (Xu et al., 2021).

2.3 Representation Learning of EHRs Data

EHRs are obtained routinely across a large number of healthcare facilities. These records are comprised of heterogeneous structured and unstructured data, including demographics, diagnoses, lab test results, prescriptions, clinical notes, images, etc. EHRs give an overview of patient health conditions from different perspectives, which enables new potential to use data-driven and machine learning methods to explore clinical events on a long-term scale (Wu et al., 2010). Diversity among EHR data is especially evident when investigating complex disorders. Nevertheless, EHR has a variety of challenging issues, including being uncurated, poor-quality, high-dimensional, sparse, heterogeneous, and incomplete (Si et al., 2021b).

For clinical problems, it is particularly crucial to build predictive models that both perform well on certain tasks, and also provide reliability and interpretability. Predictive modeling of EHR data is a machine learning task using EHR data to build a machine learning model for the purpose of predicting a certain clinical outcome of interest. The quality of EHR data representations determine how effective predictive models are for improving the performances. As a result, representation learning in EHR is a promising trend that combines large-scale data with representation learning (Bengio et al., 2013). Understanding how different physiological objects are related to each other across multiple data modalities and sequential time would construct a holistic view of patient. Representations derived from different data modalities should be constructed in a way that enables predictive models to perform effectively on many tasks.

Typically, the learning begins with extracting raw patient data (i.e., structured

or unstructured). After initial preprocessing techniques to convert raw data into numerical input embeddings, neural networks are built to derive patient representations. The networks employ either supervised or unsupervised mechanisms. The ultimate step is to evaluate how well the representations perform in predicting related clinical problems. In addition, visualizations are applied to provide some interpretations of the representations and predictions. We provide a brief overview of this learning pipeline (Figure 2.1) that is widely used in related studies. In the following subsections, we introduce the common methods of representation learning (LeCun et al., 2015) and how they are suited to encoding EHR data.

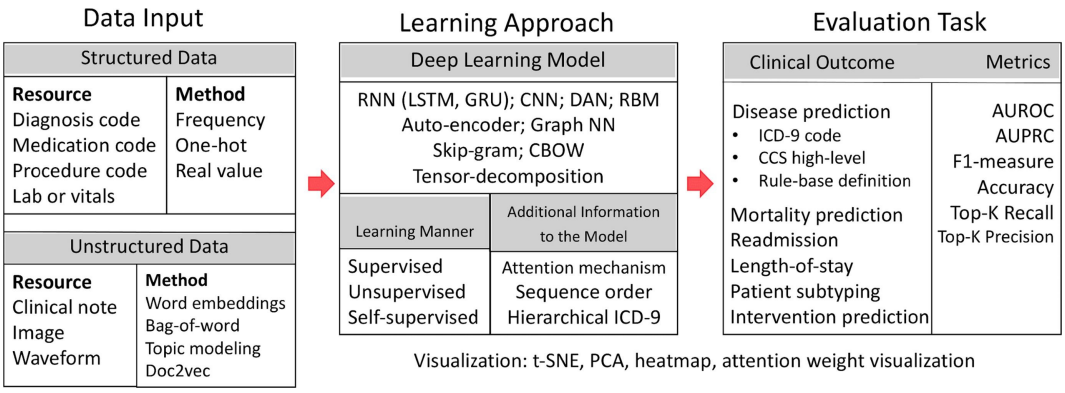

Figure 2.1: Pipeline of Patient Representation Learning

2.3.1 Temporal Matrix-based Patient Representation

The temporal matrix-based patient representation develops a two-dimensional matrix with one factor for time and another for clinical events from the EHR. The algorithm used to construct the matrix is called nonnegative matrix factorization (NMF), and it is a method for decomposing high-dimensional data into low-dimensional nonnegative components. NMF has been primarily implemented in bioinformatics

Figure 2.1 retrieved from Si et al. (2021b).

for omic data such as genetic data (Taslaman and Nilsson, 2012) and molecular profiles (Stein-O'Brien et al., 2018). Some early work also proposed performing large-scale temporal mining of longitudinal EHR data with the advanced NMF methods (Wang et al., 2012b,a; Zhou et al., 2014; Cheng et al., 2016). These early attempts have investigated the feasibility of building a mathematical two-dimensional matrix for each patient with encoded information. These works also demonstrate the challenges of processing heterogeneous EHR data as well as the potential of establishing a one-to-one defined mapping between patient and target in order to leverage the underlying knowledge of temporal EHR data.

2.3.2 Vector-based Patient Representation

Vector-based representations are commonly built with neural networks where each individual patient is represented by a low-dimensional numerical vector. Such deep neural networks include fully connected deep neural networks (fully connected DNNs), CNNs, autoencoders, and word2vec. Fully connected DNNs were inspired by neurological studies and were proposed to construct non-linear relations with one or more hidden layers (Svozil et al., 1997). In some early works, this architecture was used as a baseline method to learn patient representations (Che et al., 2015; Rajkomar et al., 2018).

CNNs were developed with modules consisting of convolutional layers and pooling layers, originally for image processing (LeCun et al., 1989). More complex variants of CNNs have been integrated to learn data types other than images, such as texts (Kim, 2014) and waveforms (Xu et al., 2018). CNNs are widely used

in multi-modal data modeling of EHRs due to their ability to process different data types (Cheng et al., 2016).

Unlike the former two networks with supervised backpropagation, autoencoders are unsupervised models that learn abstract representations through a way of reconstructing input data (Vincent et al., 2008). The success of applying autoencoders in patient representations was early proposed by Miotto et al. (2016), where a three-layer stacked denoising autoencoder was used to develop a 500-dimensional patient representation for general-purposes.

Word2vec is distinct from the above three methods in terms of learning mechanisms, where word2vec algorithms are self-supervised (Mikolov et al., 2013b). Two main algorithms, including continuous bag-of-words and skip-gram, both establish predictive relationships between target words and their surrounding contexts (Mikolov et al., 2013a). The advanced methods of word2vec have been extensively implemented to learn patient representations within clinical code sequences (Choi et al., 2016b; Xiang et al., 2019).

2.3.3 Tensor-based Patient Representation

Tensor-based patient representation learning focuses on identifying distinct patient phenotype groups and understanding the temporal variation of patients (Yang et al., 2017). The tensor decomposition and factorization method constructs a three-dimensional or more tensor aggregating clinical events for each patient. As a high-throughput extension to matrix decomposition/factorization, tensor decomposition/factorization also attempts to transform high-dimensional tensors into con-

cise low-dimensional spaces (Kolda and Bader, 2009). It effectively identifies the inherent information of high-dimensional tensors and retains the variance while decomposing them into low-dimensional factors. One of the prominent algorithms, the CANDECOMP/PARAFAC alternating Poisson regression, is widely applied for tensor decomposition or factorization by deriving a tensor as a concatenation of a fixed set of rank-one tensors (Chi and Kolda, 2012). The method is unique among traditional dimension reduction approaches in that it can leverage multi-aspect elements into different spaces and it is flexible enough to introduce domain-specific knowledge into the implementation. The superiority of tensor decomposition enables this method to learn underlying patient representations with more granularity and abstraction. More specifically, the patient tensor constructs three or more distinct components, including a patient component and other components of clinically meaningful events from diagnoses, treatments, or procedures. With a weighted combination of rank-one tensors from the product of components, each patient tensor constitutes a phenotype disease (Ho et al., 2014). Complex correlations and associations between clinical events that are not clear in flattened EHR data can be captured by a tensor-based patient representation, which is particularly useful for EHR-derived phenotype definitions (He et al., 2019).

2.3.4 Graph-based Patient Representation

Graph-based representation learning develops a concise graph to connect clinical events in EHR data, in which the nodes in the graph represent the events and the edges in-between define the associations or correlations among the events. One

of the earliest works that applied graph-based representation learning to patient EHR data was proposed by Liu et al. (2015) where they developed a temporal graph-based framework to encode the temporal relations of distinct clinical events. Graph representation learning based on neural networks known as Graph Neural Networks (GNNs) further pushed the emerging field forward. Models such as Directed Acyclic Graph (Choi et al., 2017), Graph Convolutional Network (Niepert et al., 2016), Graph Attention Network (Veličković et al., 2017), and node2vec (Grover and Leskovec, 2016) have been adapted to leverage the structure and properties of EHR data. The inherent networks with graphical structures enable a more interpretable representation. Additionally, graph representations are especially beneficial for incorporating knowledge bases into the framework. Some research has added multi-level medical code (i.e., ICD-9) knowledge graphs to the network to improve its interpretability and predictive performance (Choi et al., 2017; Ma et al., 2018). Additional knowledge base solutions include co-morbidity groups (Zhang et al., 2017) , drug-drug interactions Wang et al. (2019), omics-disease associations (Zhang et al., 2021).

2.3.5 Sequence-based Patient Representation

EHR data contains a series of unevenly distributed clinical events, and sequence-based patient representation learning attempts to learn the temporal information of these clinical events. Each patient is represented by a collection of timestamped event features. Given the limitations of conventional machine learning in dealing with challenges, state-of-the-art deep learning models for temporal EHR data rep-

resentation have been proposed, including RNNs (Sutskever et al., 2014), LSTMs (Hochreiter and Schmidhuber, 1997), and Gated Recurrent Units (GRUs) (Cho et al., 2014). These sequential neural networks are capable of processing temporal inputs. RNNs process a sequence of inputs one at a time, passing the hidden state of each unit to the next; thus, in theory, the current state contains the implicit information of the previous units (Bahdanau et al., 2014). GRUs and LSTMs are enhanced variants of RNNs to address the vanishing gradient issue of RNNs. A number of studies have used these sequential neural networks to represent patients with a sequence of clinical codes. The earliest RNN-inspired model is Doctor AI proposed by Choi et al. (2016a), in which patient trajectories are learned and represented by a skip-gram distributed vector of clinical codes.

Although RNNs have received considerable attention for sequence modeling, their limitations still exist. One significant drawback is that RNNs cannot be trained concurrently, which would increase training time. Besides, RNNs may encounter issues of vanishing or exploding gradients. More importantly, RNNs only capture inputs in a single direction, and a bi-directional RNN that learns from back and forth directions simply combines outputs from two directions, which is not considered as a true bi-directional representation. Alternatively, there is the architecture to delve deeper into the past sequences and impact future decisions, which also lets the model actively learn and determine which elements from the past are more relevant for future predictions. Thus, the Transformer architecture was proposed to mitigate the limitations of typical deep learning models and provide a stronger capability for processing sequential data compared to RNNs. The Transformer is an

encoder-decoder framework equipped with self-attention mechanisms to simulate correlations between contexts in parallel (Vaswani et al., 2017).

Thanks to the wide application of Transformer-based pre-trained models in the NLP domain (Radford et al., 2018; Devlin et al., 2019), Transformers are recently being applied for clinical sequence modeling. It also learns each clinical event as a timestamped unit and encodes the patient trajectory as a complete sequence, which is similar to how RNNs process sequence modeling of patient data. Different from the RNNs that predict the next unit in a recurrence manner, the Transformers leverage the entire sequence all together, and utilize self-attention mechanisms to capture the relevant information from it. Many recent studies have investigated the effect of pre-trained Transformers on training medical concept representations with patient EHR data (Choi et al., 2020; Song et al., 2018; Li et al., 2020; Rasmy et al., 2021).

CHAPTER 3

CLINICAL WORD REPRESENTATION

The majority of word embedding methods consider a word as a basic unit to learn the embeddings. While words, especially English words are generally made up of characters. The semantic meanings of internal characters are likewise essential to construct the semantic meanings of words. Thus, a straightforward idea is to take internal characters into consideration when generating word embeddings. We attempt to develop a joint learning model of character and word embeddings, where we learn and maintain both character and word embeddings together. They are concatenated together to generate the entire input embeddings.

Another main resource for enriching word embeddings is to make use of domain-specific corpora. Although there are a large number of general word embeddings off-the-shelf, they are mostly generated on corpora that are as large as possible so that they can cover every domain. While in the clinical domain and for clinical texts, there are generally terminologies in these specific domains. This is the case when we may have words in the corpus that would be out-of-vocabulary for general embeddings, or if the general embeddings perform poorly for the clinical NLP problem on the spot. We sought to collect a clinical domain-specific corpus and train word embeddings well-suited to clinical texts. This may also result in a common trade-off between small-but-representative corpora and large-but-not-related corpora. It is possible that the small-but-representative corpus is not broad enough

to capture all of the necessary meaning. On the other hand, the large-but-not-related corpus may not be relevant enough to adequately capture the meaning of some of the most essential terms in the corpus.

Finally, this would be a simple and general way to integrate internal character knowledge and external context knowledge to learn word embeddings that can be extended in various models and tasks.In the next few sections, we examine word-level representation learning to enhance concept extraction from clinical notes. We introduce two projects that build word representations and evaluate them to identify key terms from clinical notes. One study concentrates on cancer-related information extraction and another study focuses on public shared-task data. The goals of these two studies are to automatically identify and extract important entities from clinical notes.

3.1 Cancer-related Information Extraction

3.1.1 Introduction

Over the last decade, the amount of data kept in EHRs has increased massively. And patient data can be categorized as structured or unstructured (free-text). During the course of daily care, unstructured data such as clinical narratives, discharge summaries, laboratory reports, and pathology reports are routinely documented. As opposed to structured data, unstructured data in the EHRs typically provides richer granular and contextual information regarding clinical events while also improving communication between clinical departments. Thus, extracting information from these document resources can support a wide range of needs, ranging from clinical

decision support to secondary use of clinical data for research, as well as public health and medication management purposes.

In this section, we present an approach to extracting cancer-related information based on the notion of FrameNet. In FrameNet, frames are descriptions of circumstances established on frame semantics theory, which includes a number of participants, such as events and relations. More particularly, the terms or expressions that initiate the frame are referred to as *lexical units* (LU). The words that describe the properties or features that are associated with the LU are called frame *elements*. Considering the frame of cancer diagnosis as an example, the LUs are numerous types of cancer, such as leukemia, prostate cancer, etc. And the frame elements are cancer-related attributes that characterize a specific type of cancer, including stage, location, histology, etc. In this study, we implement a deep learning approach for identifying and extracting the lexical units and elements of the cancer frame. Deep learning models have been successfully applied to clinical natural language processing applications such as concept and relation extractions. The advantages of deep learning over traditional machine learning models (i.e., conditional random fields (CRF), support vector machines (SVM), and hidden markov models) are owing to the capability of deep learning models to train underlying features from large-scale data and consequently outperform those conventional models. Meanwhile, unlike machine learning methods that more heavily depend on human-curated features, deep learning methods rely on distributed word representations trained from large amounts of text, which would mitigate a lot of human effort while also contributing in terms of performance. Given the complexity of clinical notes, there is huge

potential for deep learning models to improve. CNN, RNN, and RNN variants such as Bi-LSTM are the most frequently utilized deep learning models for concept and relation recognition.

The remainder of this section is structured as follows. We begin by reviewing previous work, focusing on the use of deep learning methods to extract cancer-related information.Then we describe the steps in curating and annotating data, and specifically, how we define the cancer frames and elements. Next, we implement the deep neural network to extract the frames and elements in two steps. Following that, we report the experiment results. In the end, we conclude with an in-depth discussion about the current limitations and future directions.

3.1.2 Related Work

3.1.2.1 Cancer Information Extraction

A number of previous attempts have been made to extract cancer information from clinical notes. Similar research to ours have applied either rule-based methods or machine learning models or both to detect wide varieties of cancer-related information. With the increasing availability of EHR clinical notes, information extraction becomes to have an impact on hospital workflows. One of the main focuses is pathology report parsing. Previous studies have proposed NLP methods in response to different cancer types. Xu et al. (2004) investigated the capability of MedLEE (i.e., an existing NLP system) to identify tabular information relevant to cancer diagnoses from pathology reports. Weegar and Dalianis (2015) proposed a rule-based method for capturing important cancer terms from breast cancer pathol-

ogy reports. D'Avolio et al. (2008) utilized regular expressions to identify Gleason scores and TNM stages of prostate cancer from post-operative pathology notes. Ou and Patrick (2014) utilized a CRF model to identify melanoma cancer entities from pathology reports of cutaneous melanoma patients. Napolitano et al. (2010) designed a pattern-based extraction method to extract information including Gleason score, Clark level, and Breslow depth from pathology reports. Yala et al. (2017) trained a machine learning model to parse tumor characteristics from breast cancer pathology reports. Martinez and Li (2011) employed a machine learning model at the document level to parse colorectal cancer diagnosis and staging from pathology reports.

Other than pathology reports, notes such as radiology reports and MRI reports are also considered an important resource to obtain cancer-related information. Taira et al. (2001) developed an automatic structuring representation for radiologic free-text studies. Cheng et al. (2010) developed a hybrid system using an SVM model and a rule-based method to capture tumor information, including status and magnitude, from unstructured MRI reports. McCowan et al. (2007) implemented an SVM model to detect tumor, node, and metastasis (TNM) categories from the Cancer Stage Interpretation System (CSIS). Coden et al. (2009) developed an integrative cancer disease knowledge representation model (CDKRM) to localize cancer staging with the Medical Text Analysis System. Denny et al. (2012) developed a NLP system to identify colorectal cancer screening from EHR clinical notes and showed that NLP achieved higher precision while having marginally lower recall to identify patients than chart reviews.Wilson et al. (2010) detected and classified

mesothelioma patients with regard to cancer history with two rule-based methods containing Dynamic-Window and ConText. Harkema et al. (2011) investigated the positive impact of the NLP system on the quality measurement of colonoscopy.

Overall, as we have noticed, different types of cancer information overlap to a large extent across different note resources (Imler et al., 2013; Martinez et al., 2013; Ping et al., 2013; Vanderwende et al., 2013; Ashish et al., 2014; Wang et al., 2014a). A closer comparison of our study is with the work of the DeepPhe project by Savova et al. (2017), which is a document-based method to extract cancer-related entities. In this study, our method is a sentence-level approach, and we assume it would be more appropriate for clinical concepts. A frame-based approach with sentence-level information would also yield more potential for generalization across different types of data. Instead of developing task-specific NLP systems for different aims, we believe that by focusing on a frame-based approach, consistent knowledge of cancer-related information can be effectively incorporated together.

3.1.2.2 Deep Learning Models for Biomedical Texts

There has been a great deal of effort dedicated by researchers to the application of deep learning for understanding and mining biomedical text. This includes not just clinical notes, but also scientific publications, medication labels, and other types of biomedical documents. Among different models, RNN and its variants have shown outstanding results on a wide variety of concept extraction tasks. Chalapathy et al. (2016) implemented Bi-LSTM-CRF on a public concept extraction task for the 2010 i2b2 challenge and showed the superb performance of the model with

GloVe embeddings. Wu et al. (2017) compared a CNN model and an RNN model on the 2010 i2b2 challenge and demonstrated that the RNN outperformed the CNN. Liu et al. (2017b) further illustrated that the RNN achieved the best performance on three public shared tasks, including i2b2 2010 (Uzuner et al., 2011), i2b2 2012 (Sun et al., 2013), and i2b2 2014 corpora (Stubbs et al., 2015).

There are heated discussions about whether deep learning models can outperform statistical machine learning models. Habibi et al. (2017) tested both traditional CRF and Bi-LSTM-CRF on various biomedical texts and discovered that Bi-LSTM-CRF with domain-specific word embeddings could significantly improve recall while still retaining reasonable precision, resulting in an improvement in the overall F1 score. Gridach (2017) showed the efficacy and feasibility of character-level embeddings for a Bi-LSTM-CRF model.

In addition, to decide which deep learning model to use, Jagannatha and Yu (2016a) compared extensively a wide range of RNN variants, including Bi-LSTM, Bi-LSTM-CRF, Bi-LSTM-CRF with pairwise modeling, and Skip-chain CRF. The experiments on concept extraction showed that the LSTM-based variants outperformed other alternatives, in particular in parsing intricate expressions such as medication duration or occurrences (Jagannatha and Yu, 2016b).

Furthermore, Xu et al. (2017) implemented Bi-LSTM-CRF to extract the adverse drug reactions for the 2017 TAC challenge (Demner-Fushman et al., 2018) , and won the competition with the highest total score. Tao et al. (2017) trained word embeddings with MIMIC-III (Johnson et al., 2016) clinical notes and extracted the prescription information from the local hospital clinical notes. Gehrmann et al.

(2018) thoroughly compared the results of CNN and an existing NLP system, cTAKES, concluding that deep learning models like CNN should be incorporated into the system to subsequently improve the task performance. Luo et al. (2018) developed a segment-level RNN to achieve relation extractions from clinical notes, which is one of the pioneering studies on the implementation of neural networks for the classification of medical relations (Luo, 2017).

Altogether, these findings facilitate the use of deep learning models for categorizing medical concept extractions as they outperform conventional rule-based and machine learning methods while requiring minimal hand-curated feature engineering.

3.1.3 Methods

3.1.3.1 Dataset

We look into frames based on common cancer entities, including cancer diagnoses, therapeutic procedures for cancer, and descriptions of tumors, which correspond to three frames: CANCER DIAGNOSIS, CANCER THERAPEUTIC PROCEDURE, and TUMOR DESCRIPTION. A practicing physician specializing in internal medicine assists in developing a dictionary list of lexical units and correlative elements for each frame. The elements were chosen iteratively, with the main focus being on adequate frequency and importance. Table 3.1 presents the final list and corresponding descriptions. We got approval from the UT Health Institutional Review Board (the IRB protocol number is HSC-SBMI-13-0549), and extracted around 7,000 cancer-related sentences from the UT Physicians data warehouse.

Table 3.1: Frame Lexical Units and Elements

Frame Lexical Units	Frame Element	Description
CANCERMASTERFRAME		
CERTAINTY	DATETIME	POLARITY
Certainty/hedging of frame (e.g., possible, likely)	Temporal information for the frame	Existence/negation of frame
CANCERDIAGNOSIS		
adenocarcinoma, cancer, carcinoma, leukemia, lymphoma, malignancy, malignant, melanoma, myeloma, sarcoma	FAMILYHISTORY	Specifies a family member
	HISTOLOGY	Histological description
	LOCATION	Part of body
	PATIENT	Reference to the patient
	QUANTITY	Quantitative measure
	STATUS	Status (history, ongoing)
CANCERTHERAPEUTICPROCEDURE		
colectomy, hysterectomy, lymphadenectomy, mastectomy, palliative, pancreatectomy, prostatectomy, radiation, whipple	AGENT	Agent
	COMPLICATION	Unexpected outcome
	EXTENT	Extent of the procedure
	LOCATION	Part of body procedure targets
	PATIENT	Reference to the patient
	RESULT	Result of the procedure
	STATUS	Procedure status
TUMORDESCRIPTION		
lesion, mass, tumor	LOCATION	Part of body
	MALIGNANCY	benign or malignant
	MARGINSTATUS	Tumor margin
	METASTASIS	Whether has metastasized
	PATIENT	Reference to the patient
	QUANTITY	Quantitative measure
	RECURRENCE	Recurred
	RESECTABILITY	Resectable
	MORPHOLOGY	Morphology of tumor
	SIZE	Diameter/volume of tumor
	SIZETREND	The trend in tumor size
	STAGE	Stage number
	STATUS	Tumor status

We extracted sentences containing at least one potential lexical unit with keyword searching and sorted them by the TF-IDF cosine distance to improve sentence diversity and avoid using duplicate sentences. The sentences were de-identified by an automatic de-identification system and further checked by humans. The annotation was conducted on Brat (Stenetorp et al., 2012) by two student annotators and reconciled by a clinical NLP expert. For more details on data annotation, we refer you to the work by Roberts et al. (2018).

3.1.3.2 Model

In this part, we introduce the architecture of the neural network as well as the embedding approach applied to increase the model performance.

Bi-LSTM-CRF Networks: the Bi-LSTM-CRF networks was proposed by Lample et al. (2016). The LSTM model is one of the RNN variants to process long-term information and minimize vanishing gradient issues. In addition to the forward sequence information (first word to final word in the sequence), adding a second LSTM network that analyzes the identical sequence in the opposite direction (last word to first) should catch both previous and upcoming inputs, which was introduced as bidirectional LSTM (Bi-LSTM). However, the network still predicts independently at token-level labels in the decoding classification, which is not best-tailored for concept or relation extraction. Instead of token-level decoding, a linear-chain conditional random field (CRF) method was implemented in a Viterbi-style algorithm at the decoding prediction to capture the correlations and jointly produce the entire outputs of the phrases, where a state transition matrix score was calcu-

lated. The transition score and the final outputs of Bi-LSTM networks are combined to output the final probability for the prediction.

Embeddings: The vocabulary and morphology of medical terms and phrases are complicated. By incorporating character embeddings to capture morphology from medical terms, we could fully resolve the out-of-vocabulary issue and mitigate the deficiency of word-level embeddings. Similarly, we apply a Bi-LSTM model and concatenate the forward and backward output vectors to obtain the final character embedding vectors. Apart from the character embeddings, we pre-train word embeddings from clinical notes of MIMIC, which have been demonstrated to significantly improve performance in a variety of clinical NLP tasks. Because word embeddings differ greatly between general and specific domains, to balance the trade-off between small-but-representative and large-but-not-related, we experimented with two word embeddings with different settings. In the end, for input embeddings, we take full advantage of representation information for each word by feeding both character embeddings and pre-trained word embeddings into the neural network.

NLP Architecture: Our pipeline of concept and relation extraction is a two-step sequence labeling system. As shown in Figure 3.1, the first step in the pipeline is a typical named entity recognition module that extracts all the frame-trigger lexical units in sentences. In the exemplar sentence in Figure 3.1, this step extracts three lexical units as follows: *cancer* $\rightarrow$ B-CANCERDIAGNOSIS, *prostatectomy* $\rightarrow$ B-CANCERTHERAPEUTICPROCEDURE, *melanoma* $\rightarrow$ B-CANCERDIAGNOSIS.

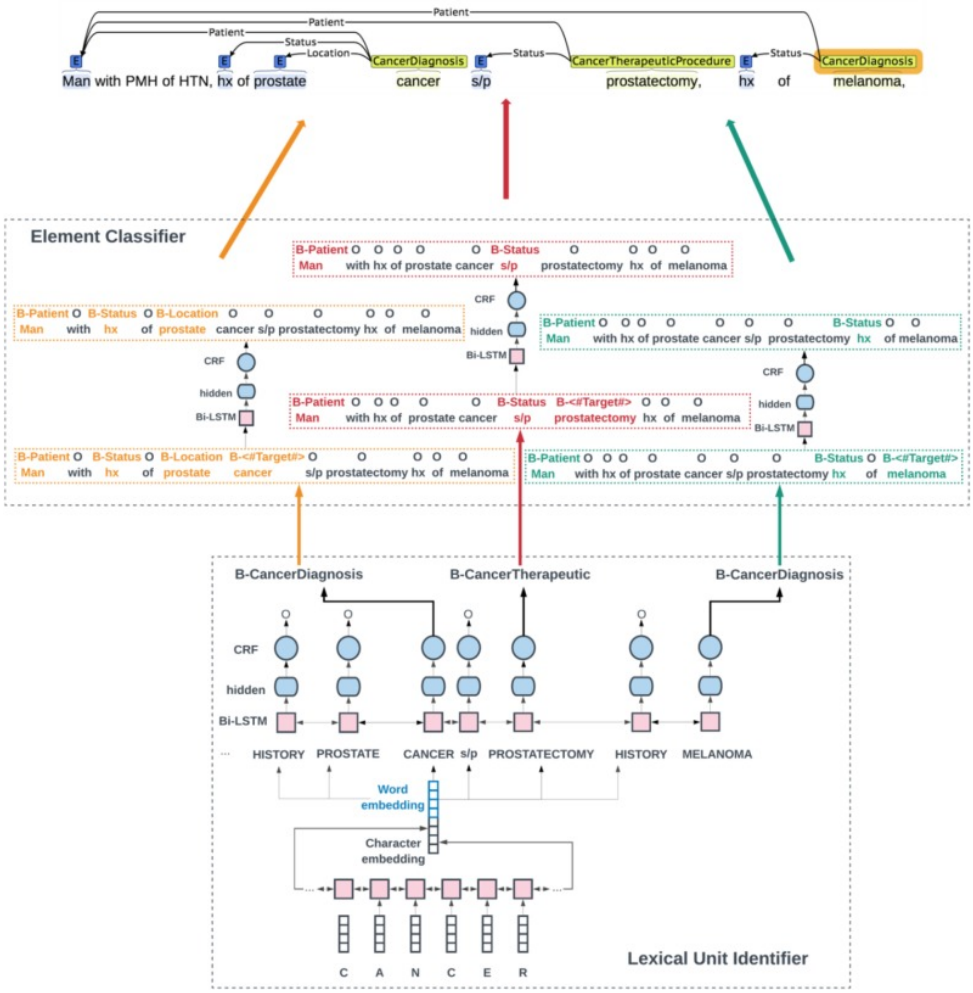

Figure 3.1: Main Architecture of the System.

Figure 3.1 retrieved from Si and Roberts (2018).

In the second step, we assign the lexical units of interest for each sentence a new label as "B-#Target#"to let the training be aware of which term is the lexical unit that triggers the current sentence. Next, the sentences are split into multiple instances, and each instance contains only one lexical unit associated with the elements. In the exemplar sentence in Figure 3.1, *cancer*, *prostatectomy*, and *melanoma* are assigned with "B-#Target#". In instance 1, the related elements of the *cancer* entity include *man*, *hx*, and *prostate*, and they are labeled as different categories from the element lists, namely B-PATIENT, B-STATUS, and B-LOCATION, respectively. In instance 1, any other lexical units and elements are labeled. Consequently, we highlight the importance of position information for the lexical unit to the element extraction in order to prevent getting confused with multiple lexical units. The final step is to combine different instances into the final result. Such a pipeline enables the architecture to incorporate all cancer information into one end-to-end framework. At execution, when a new sentence is processed, the first step extracts all the lexical units included in the sentence and, next, forwards the outputs of the lexical units to the second step, which would further extract all related elements.

3.1.3.3 Experiments

We divide the annotated sentences into three subsets, including training, test, and validation sets, with a ratio of 0.8, 0.1, and 0.1, respectively. The descriptive statistics of sample size is shown in Table 3.2. The input embeddings contain character embeddings with 100 dimensions and word embeddings. We employ a variety

of word embeddings, including 300-dimension GloVe, 100-dimension embeddings from the clinical domain, and a concatenation of both. The GloVe embeddings were released by Pennington et al. (2014) and the team trained on 6-billion words from Wikipedia. For clinical domain word embeddings, we use the embeddings trained from MIMIC-III clinical notes (Johnson et al., 2016) in a previous work (Roberts, 2016). Additionally, we experimented with the concatenation of GloVe and MIMIC-III embeddings (i.e., 400-dimension).

Table 3.2: Descriptive Statistics of Concepts

Type	**Train size**	**Dev size**	**Test size**
Sentences	6,096	762	763
Frame Lexical Unit	5,696	708	759
Frame Element	9,108	1,092	1,178

Throughout the training process, sentences and labels are segmented into two input data lists. One input list consists of sentences with solely annotated lexical units, which are processed in the first step of the NLP system to conduct lexical unit extraction. Another input list contains multiple separated sentences with labels, and each sentence has only one lexical unit with its associated elements. This input list is the input for the second step of the system, where the elements of the target lexical unit are extracted. In this way, we transform the relation extraction into a NER problem: the system classifies the relations between the potential elements and the lexical units by simply extracting the elements from each lexical-unit-targeted sentence. The gold-standard lexical unit is provided in this step.

The final training system is implemented in TensorFlow, and the model is trained on an NVidia Tesla GPU. The hyperparameters of the Bi-LSTM model are defined

as follows: hidden unit as 400 dimensions, dropout at 0.5, learning rate at 1e-4, learning rate decay at 0.99, and Adam optimization. In the second step, we also provide a 5-dimension vector embedding for the lexical unit label ("B-#Target# ").

For evaluation, we compare the prediction with the gold-standard annotations on the test set. The micro-averaged precision, recall, and F1-score for exact match are calculated. In particular, for the first step, we introduce a baseline that simply detects the existing lexical units as the real frame-trigger lexical units. This baseline has a 100% recall.

3.1.4 Results

Table 3.3 shows the performance comparison of the first step, the lexical unit identification. We report the evaluation results of four settings: the simple baseline and three embedding initializations. The embedding concatenation achieves the best overall performance across all three frames. The GloVe embeddings perform slightly better than MIMIC embeddings. The best F1-scores for CANCERDIAGNOSIS, CANCERTHERAPEUTIC, and TUMORDESCRIPTION reach 93.70%, 96.33% and 87.18% respectively. In terms of CANCERDIAGNOSIS, the F1-scores of the three embeddings are slightly the same (93%), while the combined-embedding method improves 10.72% over the baseline. TUMORDESCRIPTION gets a very low baseline with an F1-score of 79.14%, and the combined-embedding boosts it by 8.04% over the baseline.

The second step, the element classification, is calculated using the same evaluation metrics. Table 3.4 shows the overall performance results, and Table 3.5 reports

Table 3.3: Performance of System for Frame Identification

Metrics	Method	CANCER DIAGNOSIS	CANCER THERAPEUTIC	TUMOR DESCRIPTION
Precision	Baseline	70.91	91.35	65.48
	GloVe	95.35	95.88	86.72
	MIMIC-III	93.65	94.40	79.87
	GloVe+MIMIC-III	92.70	94.40	78.81
Recall	Baseline	100	100	100
	GloVe	91.99	95.88	86.05
	MIMIC-III	93.55	97.12	92.25
	GloVe+MIMIC-III	93.59	97.12	92.25
F1	Baseline	82.98	95.48	79.14
	GloVe	93.64	95.88	86.38
	MIMIC-III	93.60	95.74	85.61
	GloVe+MIMIC-III	**93.70**	**96.33**	**87.18**

the results per element type. As shown in Table 3.4, the combined-embedding method outperforms other alternatives with the best F1-score of 75.81%. GloVe embeddings perform slightly better than MIMIC-III embeddings, with an improvement of 1.64% in F1-score.

Table 3.4: General Performance Evaluation of System for Element Classifier

Embedding type	Accuracy	Precision	Recall	F1
GloVe	94.73	77.39	73.83	75.57
MIMIC-III	93.99	70.54	77.66	73.93
GloVe+MIMIC-III	94.52	73.91	77.81	**75.81**

In addition, we calculate the performance per type shown in Table 3.5. We notice that the overall result of the second step of element classification is worse than the first step of lexical unit identification. Combined-embedding outperforms in most types: MALIGNANCY: 81.82%, FAMILYHISTORY: 81.48%, DATETIME: 68.29%, RESECTABILITY: 54.55%, COMPLICATION: 52.63%, CERTAINTY: 50.87%,

Table 3.5: Cross-Frame Per Category Performance Evaluation of System for Element Classifier

	Precision			**Recall**			**F1**		
	GloVe	MIMIC	Combined	GloVe	MIMIC	Combined	GloVe	MIMIC	Combined
EXTENT	90.91	88.24	90.91	95.24	95.24	95.24	**93.02**	91.60	**93.02**
STAGE	72.34	67.31	64.81	75.56	77.78	77.78	**73.91**	72.16	70.71
PATIENT	83.75	79.21	81.91	87.78	88.89	85.56	**85.71**	83.77	83.70
HISTOLOGY	69.39	62.75	68.09	85.00	80.00	80.00	76.40	70.33	73.56
MALIGNANCY	82.35	85.00	81.82	63.64	77.27	81.82	71.79	80.95	**81.82**
LOCATION	79.69	73.55	79.13	76.40	78.76	80.53	78.01	76.07	79.82
FAMILYHISTORY	75.00	74.12	76.74	78.95	82.89	86.84	76.92	78.26	**81.48**
STATUS	78.76	71.64	71.60	73.86	79.67	76.35	**76.23**	75.44	73.90
SIZE	66.67	57.89	61.11	71.43	78.57	78.57	**68.97**	66.67	68.75
POLARITY	70.73	54.90	65.91	69.05	66.67	69.05	**69.88**	60.22	67.44
RESECTABILITY	46.15	35.00	45.00	46.15	53.85	69.23	46.15	42.42	**54.55**
CERTAINTY	59.65	45.56	48.89	40.96	49.40	53.01	48.57	47.4	**50.87**
DATETIME	23.08	50.00	66.67	15.00	60.00	70.00	18.18	54.55	**68.29**
COMPLICATION	100.000	55.56	83.33	23.08	38.46	38.46	37.50	45.45	**52.63**
MORPHOLOGY	30.00	50.00	28.57	33.33	33.33	22.22	31.58	**40.00**	25.00
SIZETREND	50.00	20.00	37.50	16.67	16.67	50.00	25.00	18.18	**42.86**
RECURRENCE	0	50.00	100.00	0	25.00	25.00	0	33.33	**40.00**
AGENT	0	0	0	0	0	0	0	0	0
MARGINSTATUS	0	0	0	0	0	0	0	0	0
QUANTITY	0	0	0	0	0	0	0	0	0
RESULT	0	0	0	0	0	0	0	0	0

SIZETREND: 42.86%, RECURRENCE: 40%. GloVe ranks second, followed by MIMIC-III in the other element types. This is reasonable because GloVe was trained from web texts in general topics, and it gets better performance in general types such as STAGE (73.91%), PATIENT (85.71%), STATUS (76.23%), SIZE (68.97%), POLARITY (69.88%). MIMIC-III embeddings only get the best performance in one type: MORPHOLOGY (40.00%), and we assume this is because MIMIC-III notes are only from Intensive Care Unit patients, which are different from cancer patient records. However, adding MIMIC-III embeddings to the input representation brings a positive impact on the representation. We observe that for some types (COMPLICATION, DATETIME, MALIGNANCY, RESECTABILITY, RE-

CURRENCE, SIZETREND), there are up to 20% improvements by adding MIMIC-III embeddings.

3.1.5 Discussion

In this work, we propose a frame-evoked NLP system based on a deep learning model to identify cancer entities and characteristics from the local hospital clinical notes. The system gets superb performance compared to the simple baseline. We implement the state-of-the-art deep learning model, Bi-LSTM-CRF, to achieve concept extraction and initialize the network with both character and word embeddings. We compare three different settings for word embeddings in both steps of the system. As a result, the overall performance of the combined-embedding method is generally higher than the methods based on individual resources.

We perform error analysis to deeply understand the system, and the result from error analysis indicates a number of commonalities. Notably, we find that the frame elements are sometimes identified for the incorrect frame. For example, *excise* (a RESECTABILITY element) belongs to TUMOR DESCRIPTION frame, but it is often identified as belonging to the frame of CANCER DIAGNOSIS. We assume this may be due to training one model on all frames. Although the information about lexical units is provided in the second step, the classifier still fails to capture the information about what the current frame is. As a result, such errors may still exist. However, the alternative way of training an exclusive model for each frame may still not solve the issue, as this would reduce the sample size significantly. Therefore, there is a well-known trade-off between sample size and an exclusive model. Due to the fact that

the frames overlap some of the frame elements, training them together substantially increases the sample size, but this also results in the type of errors described above. One solution would be to introduce the multi-task learning method, in which all three frames are learned in parallel and share parameters for some layers while also preserving task-specific portions of the model.

Also, we find out from basic knowledge in machine learning that the performance is variable corresponding to the sample size for both lexical units and frame elements. This is most evident in the false negative cases, as those cases are mainly because the samples are too limited to be identified by the classifier. For instance, TUMOR DESCRIPTION has the smallest training size, and it also gets the worst F1-score performance compared to the other two frames. This is the same with frame elements. The elements that get only 0% in F1-score are also extremely rare in the training set (37 RESULT, 18 MARGINSTATUS, 16 QUANTITY, 6 AGENT).

As discussed in the error analysis, the current limitation of this work is that we only evaluate each step in the system. In the future, we will optimize the system in a multi-task learning scenario. We will train an end-to-end model to jointly learn both lexical units and frame elements altogether. This would potentially achieve both classifications in one step, and hopefully improve the performance with the knowledge sharing. Such methods have been proposed for building an end-to-end model to jointly achieve concept and relation extraction (Miwa and Bansal, 2016; Li et al., 2017). Therefore, this would be a promising future project for further improving the performance, especially for element extractions.

3.2 Concept Extraction with Contextual Embeddings

As introduced in the previous section, word representations built with deep learning models have significantly improved the performance of many clinical NLP tasks, such as clinical concept extraction. Meanwhile, more language model-based representations have further advanced the state-of-the-art in the general NLP domain. In spite of this, there are no widely accepted practices for applying these language model-based representations into clinical NLP tasks. This section attempts to explore the range of feasible potentials for applying these new representations for clinical concept extraction by comparing them to typical word embeddings and to draw conclusions from the experiments.

3.2.1 Introduction

Clinical concept extraction is the most fundamental clinical NLP task and serves as a prerequisite to other NLP tasks, including relation extraction, co-reference, parsing, and high-throughput phenotyping. Meanwhile, language model-based representations continue to make significant progress in a wide variety of NLP tasks, ranging from natural language understanding to natural language generation. For instance, contextual representations from models including ELMo and BERT have further improved the performance of general NLP tasks. A lot of recent studies show that such representations outperform typical word embeddings in nearly all tasks.

In this section, we intend to find out what kind of impact these representations could have on clinical concept extraction. Our contributions are as follows: First,

we evaluate numerous existing embedding methods, including word2vec, GloVe, fastText, ELMo, and BERT, on four publicly-shared concept extraction tasks, which indicates the generalization of these methods. Additionally, we show the pre-training effect on pre-trained corpora and introduce the trade-off in pre-training on clinical corpora and open-domain corpora. To the best of our knowledge, this work is one of the first attempts to apply language model-based representations to clinical concept extraction, and it achieves state-of-the-art results across the board.

3.2.2 Background

In this subsection, we describe the concept of the transition from word embeddings to language model-based embeddings.

3.2.2.1 Word Embeddings

Word embeddings typically learn a dense vector with low dimensions to represent an individual word. Word2vec is one of the most well-known word embeddings, and it has been widely developed for achieving superior performance in a variety of clinical NLP tasks (Mikolov et al., 2013b). GloVe is the second word embedding approach with self-supervised learning Pennington et al. (2014). The main difference between word2vec and GloVe is that GloVe is a statistical method, and the training of GloVe relies on a co-occurrence frequency matrix. fastText is another established method for word embeddings that uses additional information such as character n-grams to mitigate out-of-vocabulary issues (Bojanowski et al., 2016).

3.2.2.2 Language Model-based Embeddings

However, the fact that word-level embeddings conflate all the different semantic meanings of a word limits their effectiveness, and the embeddings do not adapt to the context. In order to address these shortcomings, advanced methods have tried to explicitly encode the surrounding context of words into the word representation, which has proven successful.

The first contextual word representation we evaluate is ELMo, proposed by Peters et al. (2018). As opposed to typical word embeddings, which create a single vector for a word and keep it unchanged in NLP tasks, ELMo captures and dynamically modifies the word embeddings through a multilayer representation. Before actually working on some NLP tasks, the model first learns the contextualized information from a large-scale text corpus, which is known as the pre-training step. Following this, the inner states of the pre-trained language model are fed into the actual NLP tasks, which are considered the context-sensitive word embeddings. By adding weights to the inner states and optimizing the weights towards the loss of the downstream task, the word representations become more well-suited to the surrounding context in the corpus. Thus, the downstream NLP task would obtain a decent initialization and achieve optimal performance.

Another more advanced language model-based word representation is BERT, proposed by Devlin et al. (2019), which also starts from pre-training on a large-scale unlabeled corpus. Unlike ELMo, which applies layers of inner states, BERT adopts a model-wise manner to encode context information in sentences with a fully trainable bidirectional transformer. The transformer (Vaswani et al., 2017) is a mul-

tiheaded self-attention mechanism with position embeddings. More importantly, regarding the approach of how to utilize those representations in the downstream NLP tasks, ELMo is a feature-based extraction approach and BERT is a fine-tuning method. The feature-based extraction is the same with word-level representation that extracts layers of inner-state vectors of the model and uses the vectors as the input embeddings. While the fine-tuning method dynamically alters the entire language model and tailors the model to the downstream task, which results in a task-specific fine-tuning model. During the fine-tuning, the BERT model is entirely fed into the target task, which is assumed to contain more context information and is more likely to achieve good prediction results.

3.2.2.3 Clinical Concept Extraction

Clinical concept extraction is the process of extracting clinical entities from clinical notes and analyzing them, which is usually considered a sequence labeling problem to be tackled with machine learning–based models. Deep learning–based models with word embeddings as the input features have recently been shown to be effective. The current state-of-the-art model for clinical concept extraction is the Bi-LSTM-CRF model as introduced in the previous section. The bidirectional recurrent neural network captures both forward and backward input in the sentence, and the CRF layer before the classification uses the Viterbi algorithm to decode sequential output patterns. Similar to this work, a few recent studies also investigated contextual representations for extracting entities from biomedical documents. For example, Zhu et al. (2018) improved clinical concept extraction of the i2b2

2010 dataset with ELMo. Lee et al. (2020) mainly applied BERT to concept extraction from biomedical literature, and they also pre-trained BERT models from PubMed literature, named BioBERT. A closer benchmarking to our work is proposed by Alsentzer et al. (2019), where the authors pre-trained MIMIC-III clinical notes (Johnson et al., 2016), but gets lower prediction performances on two corpus in common: the i2b2 2010 and 2012 datasets. Their findings show that only pre-training on discharge summaries, rather than on all notes, would be beneficial, and they also continued the pre-training with literature corpus. There are also a number of studies that apply BERT to clinical prediction tasks such as 30-day readmission prediction (Huang et al., 2019), and other standard clinical outcomes of interest and NLP tasks (Peng et al., 2019).

3.2.3 Methods

3.2.3.1 Embeddings

We experiment with both released embeddings from the open domain and pre-training embeddings from MIMIC-III clinical notes (Johnson et al., 2016). For the traditional word embeddings, we apply the Bi-LSTM CRF to achieve the concept extraction and feed the fixed embeddings into the model. For pre-training, we set the minimum frequency of words as five, which means that words that appear at least five times are included, and the rest are denoted as "UNK ". Character embeddings for each word are also being considered to mitigate the out-of-vocabulary issue.

Regarding ELMo embeddings, to create context-sensitive embeddings, context-

independent embeddings are assigned with trainable parameters, which are further input into the target task. The context-sensitive embeddings are achieved by a low-dimensional deduction with a highway connection followed by a stacked multi-layer. The stacked layer originally comes from the ELMo architecture and includes a character-based CNN and a 2-layer Bi-LSTM language model (i.e., Bi-LM). So the context-sensitive embeddings consist of a trainable combination of highway connections of the Bi-LM. Because ELMo already incorporates character embeddings with char-CNN, the use of character embeddings for the downstream tasks is not necessary. In the end, the context-sensitive embeddings of ELMo are extracted and fed into the state-of-the-art concept extraction model (i.e., Bi-LSTM CRF) for downstream NLP predictions.

In terms of embeddings from BERT models, we first download both BERT_{base} and BERT_{large} off-the-shelf: $\text{BERT}_{base}(\texttt{General})$ and $\text{BERT}_{large}(\texttt{General})$. The former has 110 million total parameters and the latter has 340 million. The detailed architecture of these two models can be referred to in Devlin et al. (2019). Compared to BERT_{base}, BERT_{large} is deeper in terms of the network with 24 layers of transformer encoders, each of which has 1024 hidden units and 16 attention heads. Otherwise, these two models essentially have the same networks. In fine-tuning, the downstream tasks (i.e., clinical concept extractions) are initialized with the parameters from $\text{BERT}_{base}(\texttt{General})$ and $\text{BERT}_{large}(\texttt{General})$. Because BERT already incorporates enough label-correlation information, the CRF layer is removed and the concept extraction is achieved only with a Bi-LSTM layer at the final classification. In addition to the two BERT(General) mod-

els, we also pre-train clinical BERT models with MIMIC-III clinical notes following the pre-training instructions. The pre-training starts from BERT$_{base}$ and BERT$_{large}$ checkpoints, and is further tailored to MIMIC-III clinical notes. We refer these two clinical domain specific pre-trained models as BERT$_{base}$(MIMIC) and BERT$_{large}$(MIMIC).

3.2.3.2 Datasets

The concept extraction tasks in this work are four widely-studied clinical NLP tasks: the i2b2/VA 2010 challenge (Uzuner et al., 2011), the i2b2 2012 challenge (Sun et al., 2013), the SemEval 2014 Task 7 (Pradhan et al., 2014), and the SemEval 2015 Task 14 (Elhadad et al., 2015). The descriptive statistics of each corpus are shown in Table 3.6. We notice that the i2b2 2010 is much larger than the other three, but they are all relatively sizable for clinical concept extraction. For the i2b2 2010, there are three types of clinical concepts to be extracted: PROBLEM, TEST, TREATMENT. For i2b2 2012, there are 6 types: PROBLEM, TEST, TREATMENT, CLINICALDEPARTMENT, EVIDENTIAL, OCCURRENCE. There are only one concept type for the SemEval 2014 and 2015 : DISEASEDISORDER, which is similar to PROBLEM in the two i2b2 tasks that describe the specific disease.

For the pre-training datasets, as described above, we used all MIMIC-III clinical notes, which include nearly 2 million notes. After some basic cleaning, we get 1,908,359 notes with 786,414,528 tokens and a vocabulary size of 712,286. Words are lower-cased in pre-training traditional word embeddings but not in pre-training ELMo and BERT.

Table 3.6: Descriptive Statistics of Concepts

Dataset	Subset	Notes	Concepts
i2b2 2010	Train	349	27,837
	Test	477	45,009
i2b2 2012	Train	190	16,468
	Test	120	13,594
SemEval 2014 Task 7	Train	199	5,816
	Test	99	5,351
SemEval 2015 Task 14	Train	298	11,167
	Test	133	7,998

3.2.3.3 Experiment Details

For concept extraction with regards to Bi-LSTM-CRF settings, we use the following: 512 hidden units, a dropout of 0.5, a learning rate of 0.001 with a decay of 0.9, and Adam optimization. All experiments remain the same in terms of the Bi-LSTM-CRF hyper-parameters.

For pre-training, across all embedding methods, two pre-training scenarios are experimented with: the first is to use an off-the-shelf released model (`General`), and the second is to pre-train further on MIMIC-III clinical notes (`MIMIC`). For the first scenario, the details about open source models are shown in Table **??**. We also experiment with BioBERT, which is pre-trained with PubMed biomedical literature and initiated from the BERT$_{base}$.

For the second scenario, the dimension of word embeddings is 300 across all traditional embedding methods to equalize the off-the-shelf embeddings. We define the hyper-parameters for training word embeddings as follows: window size: 15, word count: 5, iterations: 15, embedding dimension: 300.

For ELMo, we follow the instructions from Peters et al. (2018) to set up the

Table 3.7: Resources of Off-the-shelf Embeddings

Method	**Resource (#.Tokens/ #.Vocabs)**	**Size**	**Language model**
GloVe	Gigaword5 + Wikipedia2014 (6B/ 0.4M)	300	NA
fastText	Wikipedia 2017+ UMBC webbase corpus and statmt.org news (16B/ 1M)	300	NA
ELMo	WMT 2008-2012 + Wikipedia (5.5B / 0.7M)	512	2-layer, 4096-hidden, 93.6M parameters
BERT_{base}	BooksCorpus+ English Wikipedia (3.3B/ 0.03M*)	768	12-layer, 768-hidden, 12-heads, 110M parameters
BERT_{large}	BooksCorpus + English Wikipedia (3.3B/ 0.03M*)	1024	24-layer, 1024-hidden, 16-heads, 340M parameters

hyper-parameters. MIMIC-III is divided into a pretraining corpus with a ratio of 0.8 and an evaluating corpus with a ratio of 0.2 for reporting perplexity. The pretraining step has 15 iterations, resulting in an average perplexity of 9.929.

For BERT, we also follow the default settings to set up the pre-training. The vocabulary list applies to the released list with 28,996 word-pieced tokens. We save the intermediate model at each twenty thousand steps and evaluate the performance of intermediate checkpoints.

In terms of the fine-tuning, we apply an adjustment with Xavier initialization instead of random initialization on the Bi-LSTM layers. Then an early stop mechanism of 800 steps is launched to prevent over-fitting. Lastly, the outputs from BERT are still wordpieced-level, which requires post-processing to align with the gold standard concepts.

Overall, 10% of the training set is split into development sets and the official

test set is kept to report the performance. The performance metrics are precision, recall, and F1-score for exact matching. The experiments are implemented with TensorFlow on the NVidia Tesla V100 GPU (32G).

3.2.4 Results

3.2.4.1 Performance Comparison

The performance of different embedding approaches on the four clinical concept extractions is evaluated, as shown in the Table 3.8. On the first note, embeddings pretrained on the clinical corpus outperform those off-the-shelf models. The `General` embeddings outperform the `MIMIC` embeddings with a relatively high increase. The best result for the i2b2 2010 task gets 90.25 in F1-score with BERT$_{large}$(`MIMIC`). Compared to the best F1-score of the traditional embeddings by GloVe (`MIMIC`), it improves the F1-score by 5.18. Similarly, the best performances for the i2b2 2012 task, SemEval 2014 and 2015 task are achieved by BERT$_{large}$(`MIMIC`), with F1-scores of 80.91, 80.74 and 81.65, respectively. The current state-of-the-art performance for these four tasks is reported with an F1-score of 88.60 (Zhu et al., 2018), 92.29 (Liu et al., 2017b), 80.3 (Tang et al., 2015), and 81.3 (Zhang et al., 2014), respectively. Notably, with the clinical domain pre-trained language model, we achieve new state-of-the-art results across the entire four tasks.

3.2.4.2 Pretraining Evaluation

We investigate the effectiveness of pre-training by calculating the loss and the prediction performances on intermediate models. As shown in Figure 3.2, we notice

Table 3.8: Test Set F1 Comparisons in Exact Matching

Method	i2b2 2010		i2b2 2012		Semeval 2014		Semeval 2015	
	General	MIMIC	General	MIMIC	General	MIMIC	General	MIMIC
word2vec	80.38	84.32	71.07	75.09	72.2	77.48	73.09	76.42
GloVe	84.08	85.07	74.95	75.27	70.22	77.73	72.13	76.68
fastText	83.46	84.19	73.24	74.83	69.87	76.47	72.67	77.85
ELMo	83.83	87.8	76.61	80.5	72.27	78.58	75.15	80.46
BERT$_{base}$	84.33	89.55	76.62	80.34	76.76	80.07	77.57	80.67
BERT$_{large}$	85.48	**90.25**	78.14	**80.91**	78.75	**80.74**	77.97	**81.65**
BioBERT	84.76	-	77.77	-	77.91	-	79.97	-

that for ELMo and BERT, the training loss always drops with the steps progressing, which means that the language model is gradually adjusting to the clinical language. If there is no action to interrupt the pre-training process, the final loss value will be extremely small. But this will eventually induce overfitting on pre-training data.

We experiment with the i2b2 2010 task to report the performance at each intermediate checkpoint throughout the process. It is interesting to observe that the performance of ELMo stays stable after a specific number of rounds, reaching the best F1-score of 87.80 at step 280K. The performance of BERT$_{base}$ on the downstream task is less consistent and tends to drop after obtaining the best result (i.e., the maximum F1 is 89.55 at the 340K step). We hypothesize this is because the clinical domain pre-trained model is initialized with the off-the-shelf BERT model; after several runs on the MIMIC data, the knowledge gained from the large open-domain corpus is forgotten, and the training finally converges on a model which is nearly closer to one built from scratch. Therefore, constraining pre-training on a clinical corpus to a finite number of steps is a beneficial trade-off that leverages the capabilities of a large open-domain corpus with the effect of learning from a

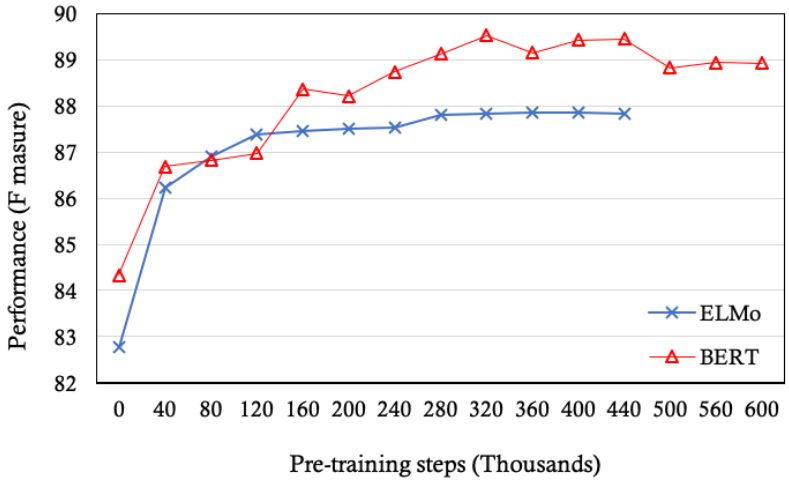

Figure 3.2: Performances on the i2b2 2010 Task by Pre-training Steps

clinical corpus. We anticipate that this will be a useful reference for clinical NLP fields when developing customized pre-trained models from the clinical corpus.

3.2.5 Discussion

In this section, we examine the impact of a variety of embedding approaches on four public clinical concept extractions. As expected, domain-specific models exceed off-the-shelf models. When pre-trained on a clinical corpus, all sorts of embeddings achieve consistent improvements in the majority of tasks. Additionally, contextual embeddings outperform traditional embeddings in predictions. Thus, we conclude that significant improvements can be obtained by pre-training a deep language model on a large corpus and then specifically adapting it for the downstream tasks.

Figure 3.2 retrieved from Si et al. (2019).

One significant distinction between BERT and other embeddings is that BERT dissects words into subwords known as word-pieced tokens.Instead of relying on a morphology lexicon, this is performed by statistical inference of a large corpus. Thus, the question for clinical NLP is whether a different way of tokenizing is applicable for clinical language as compared to general text (such as Wikipedia and web pages). We look into the word-pieced tokens for clinical terms from the well-known lexical similarity corpus proposed by Pakhomov et al. (2016). As predicted, the findings do not exactly match the morphology of standard medical terms. For instance, *appendicitis* in word-pieced tokens, is *app*, *-end*, *-icit*, *-is*, in contrast to the medical suffix of *-itis*. This does not necessarily imply poor segmentation, and it is feasible that this might perform better than a word-piece based on the SPECIALIST lexicon (Browne et al., 2000).

One idea for further exploration is to develop word pieces from MIMIC-III. This is not as straightforward as it appears at first glance. We have a major issue because the BERT models we are using in this study were first pre-trained on a 3.3 billion open-domain corpus before being pre-trained even further on MIMIC-III. In order to get similar language models while conducting word-pieced tokenization on MIMIC-III, it would be necessary to replicate the pre-training on the open-domain corpus at least. We leave this exploration to future work due to the large number of experiments to be evaluated so as to establish the optimal word-pieced strategy.

CHAPTER 4

CLINICAL NOTE REPRESENTATION

In this chapter, we will investigate the document representation for clinical notes. More specifically, we attempt to implement state-of-the-art deep learning models for processing long and multiple clinical documents altogether. We consider the complexities of processing clinical notes and develop models to encode the characteristics of clinical notes, including their hierarchical, longitudinal, and contextual characteristics. In the next section, we first introduce the motivation for these complexities, and for each complexity, we propose neural networks that are appropriate for encoding sufficient information in the model for processing large-scale clinical notes.

4.1 Hierarchical Convolutional Neural Network

4.1.1 Motivation

Words, sentences, and documents are all essential linguistic stacks of natural language. The majority of models for learning clinical texts are only devoted to training words directly towards labels, which fails to incorporate the hierarchical structure hidden in the clinical texts from words, sentences, documents, and eventually to labels. Similarly, like from characters to words, it is also important to consider the hierarchical structure between different language hierarchies and encode them into the model. This is essentially achieved by pooling the outputs from the previous level and using this pooling as the input for the next level.

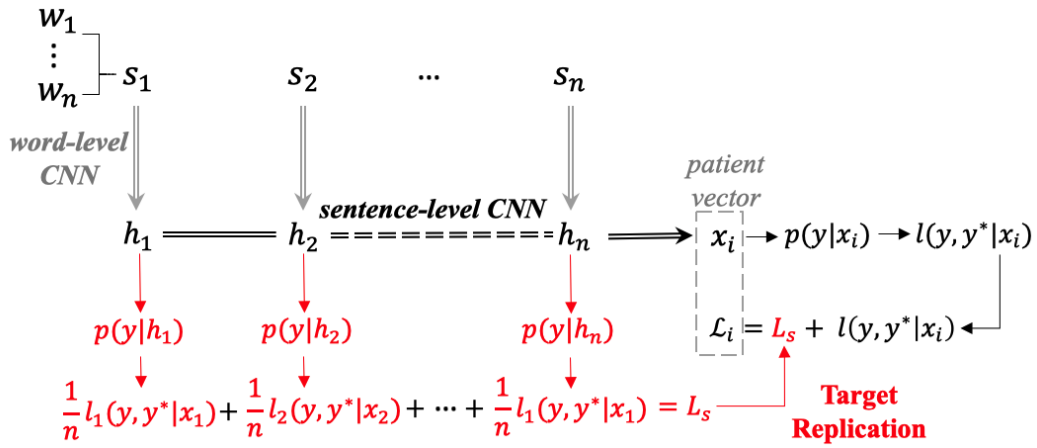

Figure 4.1: Hierarchical Convolutional Neural Network Model Architecture

4.1.2 Model Architecture

Considering that, we develop a two-level CNN, including a word-level CNN and a sentence-level CNN. The network architecture is shown in Figure 4.1. Specifically, the word embeddings are assembled to obtain a single sentence embedding (i.e., word-level CNN). The sentence embeddings are aggregated to form the patient representation (i.e., sentence-level CNN). At this time, because we simply combine many notes of a single patient into one document, the model does not take temporal information into account. But we still want to emphasize the relations between sentences, so we implement the target replication. That means, for each sentence from the word-level CNN, we also calculate the loss of each sentence towards the label, and then add the sum of these losses from individual sentences to the final loss at the final prediction. For the prediction, we utilize the softmax function before the final classification, and the prediction is optimized with a cross-entropy loss.

4.2 Hierarchical Attention Network

4.2.1 Motivation

Due to the variable distribution of EHR records, addressing temporal sequences of patient data is complicated yet desirable to enhance predictive performance. Recent progress in sequential deep neural networks, such as the variants of RNNs, has demonstrated superior results for representing sequential EHR data. However, limited work on clinical note modeling has taken into account temporal relations between notes. In particular, we assume that the sequence orders associated with clinical notes may not always correspond to the temporal reality in clinical settings. Clinical notes collected over a long period of time contain significant temporal dependencies, such as progress notes for a single patient on different dates. Clinical notes in small periods are either generated in bursts or from different units, solely presenting patients from different aspects, resulting in less temporal information.

To overcome these challenges, we develop a three-level HAN for the purpose of learning clinical texts. The HAN network adopts RNNs at three hierarchical levels, each of which comprises encoders and attention mechanisms. The model is mainly composed of words in a sentence, sentences in a document, and documents (or sets of documents) in a patient. At each layer, attention mechanisms are used to determine which information is the most useful in prediction. Additionally, we greedily separate documents at the patient level into sets of documents dependent on time frames. In the next subsection, we introduce the details of the proposed model.

Yang et al. (2016) first proposed the HAN as a hierarchical model for document

classification. Because of its capacity to deal with large amounts of text, HAN has been extensively used in clinical NLP tasks such as classifying patient safety (Cohan et al., 2017a,b), assigning diagnostic codes (Samonte et al., 2018, 2017; Mullenbach et al., 2018; Baumel et al., 2018; Du et al., 2019), analyzing radiological reports (Banerjee et al., 2019), capturing cancer entities (Gao et al., 2017), and forecasting mental illnesses (Tran and Kavuluru, 2017; Ive et al., 2018). Recent developments have also developed advanced hierarchical networks that use structured EHR data to make clinical predictions.

Although more advanced pre-trained language models such as BERT and XL-Net now achieve state-of-the-art results in many NLP tasks, it remains unclear how these models can be used to deal with long-distance relations in clinical notes. When constructing patient representations from clinical notes, few studies have taken into account information like the time sequence of patient data and the hierarchical structure of natural language. To our knowledge, this is one of the earliest works to use a hierarchical RNN and attention mechanism for specifically processing longitudinal clinical notes while simultaneously incorporating hierarchical information from free text.

4.2.2 Model Architecture

Figure 4.2 shows the network architecture. The network constructs the final patient representation gradually from word embeddings. Initially, to create sentence representations at the sentence level, the model recognizes significant words with higher attention weights automatically and concatenates into sentence representations.

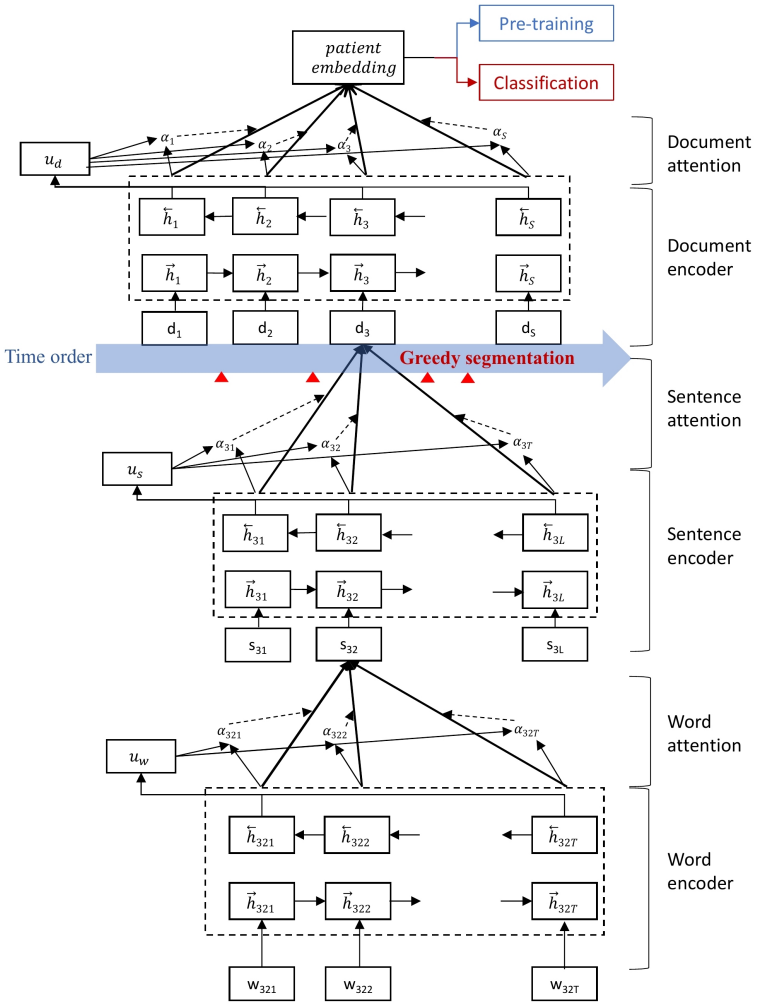

Figure 4.2: Hierarchical Attention Network Model Architecture

Figure 4.2 retrieved from Si and Roberts (2020).

To generate document representations at the document level, the model continues to apply attention mechanisms to capture the important sentences and assemble them to build meaningful document representations. Finally, the model combines document representations into a final patient representation at the patient level. Following that, the patient representations are applied directly to the outputs for prediction or pre-training purposes.

The model consists of two components at each hierarchy level: an RNN encoder and an attention mechanism. The encoder is a bidirectional RNN, which keeps the sequences in both backward and forward directions during the long sequence. The outputs of hidden units are fed into a fully-connected network with the softmax function to generate normalized attention weights; hence, the attention at each level (i.e., word, sentence, and document) symbolizes the significance of different portions.

After multiplying normalized attention weights by hidden-unit values, the weighted sum of hidden layers is computed as the final attention-weighted output. Finally, we get the patient representation, which consists of the weighted sum of document embeddings. Using this patient vector, one may obtain a hierarchical representation of the patient that can be further used for a variety of classification tasks.

4.2.3 Implementation Details

Previously, we introduced two-level CNN to integrate hierarchical information, and in this section we propose a three-level HAN to incorporate both hierarchical and longitudinal representations. We compare these two neural networks with different

levels on the patient mortality prediction tasks. Totally, we have four models: a two-level CNN, a three-level CNN, a two-level HAN, and a three-level HAN. The hyperparameters of HAN are defined as follows: RNN with a hidden unit of 200 dimensions, the attention output sizes are 300, 150, and 100 for the word, sentence, and document attentions, respectively. For CNN, the numbers of filters for words, sentences, and document levels are 100, 50, and 50, respectively. Each filter consists of a vector of [3, 4, 5]. Because the class distribution for patient mortality is not balanced, we use the area under the receiver operating characteristic curve (AUC) and the area under the precision-recall curve (AUPRC) as the performance metrics.

4.2.4 Performance Comparison

Table 4.1 shows the performance results of three mortality prediction tasks using five different methods in AUC and AUPRC.The 3-level HAN model with the best time split outperforms the alternatives in AUPRC of 78.74, 74.63, and 71.28 for the predictions of in-hospital, 30-day, and 1-year mortalities, respectively. In general, sequential networks perform better than convolutional networks on the majority of tasks when both of them are trained at the same hierarchical level, meaning that the sequence between clinical notes is essential to contributing to the mortality prediction. Also, 3-level models outperform 2-level models of the same network for nearly all predictions, demonstrating the importance of the hierarchical structure in modeling clinical notes.

Table 4.1: Performance of Different Models on Mortality Prediction Tasks

	Tasks	**In-hospital**		**30-day**		**One-year**	
MODELS		AUC	PRC	AUC	PRC	AUC	PRC
CNN	2-level	92.47	70.21	82.11	65.79	81.55	65.34
	3-level	94.04	72.78	85.76	70.27	84.31	69.35
HAN	2-level	93.96	75.13	85.30	71.28	84.62	69.14
	3-level w/o segmentation	94.06	74.24	86.15	72.35	86.88	70.96
	3-level with the best segmentation (max time span: 1-hour)	**94.92**	**78.74**	**87.59**	**74.63**	**87.41**	**71.28**

4.3 Hierarchical Transformer Network

4.3.1 Motivation

We have introduced three characteristics that are important to clinical note understanding: contextual, hierarchical and longitudinal, and we intend to consider these three altogether in one architecture. Before we describe the architecture details, we first emphasize the motivation for developing this network.

Recently, transformers are becoming more popular and perform well on numerous NLP tasks (Vaswani et al., 2017). The Transformers network completely eliminates convolution and recurrence and only depends on attention mechanisms and position embeddings. BERT, based on the Transformer, has been established as the current state-of-the-art model for many clinical NLP studies (Devlin et al., 2019). The BERT model has a limit on the length of the input texts, which restricts the applicability of parsing across a series of long documents. To address this issue, prior work has considered segmenting long texts into smaller components and then averaging their corresponding representations through BERT (Adhikari et al., 2019;

Pappagari et al., 2019). These methods, however, overlook the temporal relations between documents as well as the hierarchical structure between contexts (Yang et al., 2020). When reading a long series of documents like full-length novels, legal terms, and clinical notes, it is important for humans to understand the hierarchical and longitudinal information between contexts. This is also true for deep learning models, as the model should also integrate such information.

We propose Hierarchical Transformer Networks to capture the contextual, hierarchical, and longitudinal information from long-sequence documents, leveraging all three interrelations. BERT models are used explicitly at the word level. Different sizes of BERT models and input sequence lengths are investigated to evaluate the trade-off between model size and sequence length. At both sentence and document levels, we use transformer-based encoders. The challenge of effectively training Transformers (Popel and Bojar, 2018) necessitates extensive experiments with a wide range of hyper-parameter settings. We also enable the server with distributed training to overcome memory issues and accommodate longer input sequences.

We note that while the notion of a hierarchical network for Transformers might not be conceptually novel, the fact that it has not yet been proposed for processing long-sequence clinical notes demonstrates that there are serious challenges to such a method. The difficulties largely exist; for example, optimization failure without appropriate learning rates, convergence difficulty without valid initializations, overfitting easily on training sets without proper dropout. Our main contribution is to make the model applicable and feasible to train for long and multiple text classification, as we are not simply classifying an individual document, but rather large

collections of documents longitudinally over time (i.e., one classification for all of a patient's notes). To the best of our knowledge, this is the earliest attempt to build the Hierarchical Transformer Network for modeling long and multiple clinical text classifications. We assume this architecture essentially learns the contextual information of sentences while also leveraging longitudinal and structural information at each hierarchical level.

4.3.2 Model Architecture

The architecture of Hierarchical Transformer Networks is shown in Figure 4.3. At each level, the model automatically aggregates the full sequence with pooling into the following level, and the input length is cut or padded to a fixed value. In the following subsections, we specifically describe each model component.

4.3.2.1 Word-level BERT

At the word-level, we apply a BERT model with word-pieced tokens as the input. All parameters in the model are trainable. We retain both of the special tokens `[CLS]` and `[SEP]` at the start and the end of the sentence. `[CLS]` is the first unit of each sentence, and its hidden state is generally the representation of the entire sentence. `[SEP]` is located at the end of the sentence to distinguish between different sentences. We discard the segment embeddings but preserve the position encoding. Thus, the input embeddings of a certain token are the concatenation of word-pieced embeddings Tok_i, and position encodings P_{i}.

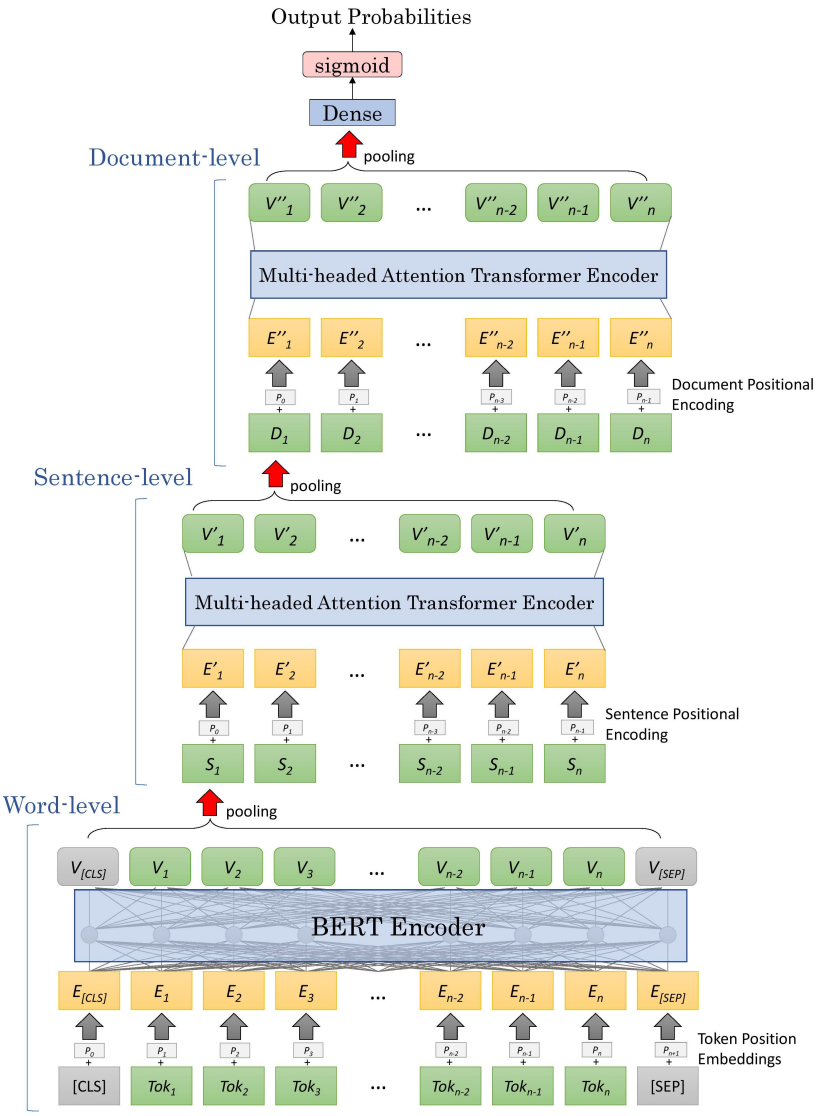

Figure 4.3: Hierarchical Transformer Network Model Architecture

Figure 4.3 retrieved from Si and Roberts (2021).

4.3.2.2 Sentence- and Document-level Transformers

Transformer-based encoders are stacked to represent sentences and documents. We recommend the work by Vaswani et al. (2017) for more details about Transformers. Briefly, a Transformer-based encoder is N layers of multi-headed self-attention and fully-connected feed forward networks. The self-attention head has three inputs – Q(query), K(key), V(value) – to pass by the scaled dot-product attention. The hidden outputs from the attention are added with a linear dense layer. Q, K, and V are partitioned into several heads to allow the model to attend to information from different representation subviews. Position encodings are also considered, as the same strategy at the word level. The input of the sentence is determined by the first `[CLS]` token, denoted as the *CLS-pooling*. We also explore different pooling methods, including *mean*, *max*, and *mean_max poolings*, for deriving the outputs from the prior levels. This allows higher-level of the model to have additional access to lower-level rather than relying just on what is encoded in the `[CLS]`. The final prediction is achieved by implementing a dense layer on the classification and using a Sigmoid function to output the probabilities. Also, the model could be widely applied to various machine learning tasks like pre-training and clustering with appropriate loss functions.

4.3.2.3 Adaptive Segmentation and Filling

As discussed in Section 4.2.1, we assume that the sequence orders associated with clinical notes may not always correspond to the temporal reality in clinical settings. Clinical notes collected over a long period of time contain significant temporal de-

pendencies, such as progress notes for a single patient on different dates. Clinical notes in small periods are either generated in bursts or from different units, solely presenting patients from different aspects, resulting in less temporal information. To distinguish short-period coincidences from long-scale dependencies, we proactively split or combine clinical notes to reflect the realistic sequence between notes. Furthermore, such a method reduces the number of documents fed into the model, allowing the model to learn long-term relationships more accurately.

Formally, for each individual, we begin by sorting the notes chronologically, and then use a greedy approach to determine the split points. The algorithm attempts to reduce the maximal time range of two adjacent groups. Given T documents in a sequence $\{d_t\}_{t=1}^{T}$, we have k-1 segmentation points $\{s_i\}_{i=1}^{k-1}$ to split the sequence into k groups $\left\{G_j\right\}_{j=1}^{k}$, where

$$G_j = \begin{cases} \{d_t \mid d_t.\text{time} < s_1\}, & \text{if j} = 1 \\ \{d_t \mid d_t.\text{time} \geq s_{k-1}\}, & \text{if j} = \text{k} \\ \left\{d_t \mid d_t.\text{time} \in \left[s_{j-1}, s_j\right)\right\}, & \text{otherwise.} \end{cases} \tag{4.1}$$

where $d_t.\text{time}$ is the charttime of document d_t. The span of a group is defined as the time difference of the earliest and the latest document in the group:

$$\text{span}\left\{G_j\right\} = \max_{d_k \in G_j} \{d_k.\text{time}\} - \min_{d_{k'} \in G_j} \{d_{k'}.\text{time}\} \tag{4.2}$$

The optimal choice of the segmentation points can be found by minimizing the

following:

$$\hat{s}_1, \dots \hat{s}_{k-1} = \underset{s_1,\dots s_{k-1}}{\operatorname{argmin}}\{k\} \tag{4.3}$$

$$\text{subject to} \quad \max_{1 \le j \le k} \left\{ \operatorname{span}\left(G_j\right) \right\} \le D \tag{4.4}$$

where D constrains the upper bounding of the span. Intuitively speaking, the notes inside a defined maximum time range are regarded as a single "note". The notes outside of the span are split into different units. By doing so, we aim to maintain the temporal relation of notes over long periods whilst also grouping notes that emerge in bursts.

4.3.3 Experiment Details

4.3.3.1 Dataset and Prediction Tasks

The experiments are conducted with patient notes from MIMIC-III (Johnson et al., 2016). The proposed model is evaluated for its ability to predict in-hospital mortality and phenotypes. Both tasks are common clinical outcomes that are important for guiding clinical decisions. Descriptive statistics of patient data are shown in Table 4.2.

In-hospital Mortality Prediction: MIMIC-III indicates the time of death for patients who die in hospital, which allows us to establish precise cohorts for in-hospital mortality. To avoid confusion with different admissions of one patient, we only consider patients with a single encounter. We exclude *discharge summaries* in the mortality prediction task because discharge summaries literally describe the mortality outcome in text. For the same reason, we also remove notes with a chart

Table 4.2: Descriptive Statistics of Datasets

		In-hospital Mortality	Phenotype Prediction
# Total Patients (% Positives)		30,881 (13.80%)	30,990 (Table 4.3)
# Notes Per Patient	Mean	18.1	16.9
	Median	12	11
	80 *%tile*	24	22
# Sentences Per Note	Mean	29.8	37.4
	Median	18	21
	80 *%tile*	42	50
# Wordpieces Per Sentence	Mean	19.2	18.9
	Median	12	12
	80 *%tile*	22	22
# Total Sentences		16,662,894	19,656,126
# Total Notes	Raw	906,717	866,735
	Adaptive	559,942	525,222

time later than the time of death and discharge time.

Phenotype Prediction: The goal of phenotype prediction is to categorize patients into a number of phenotypes. We chose the top ten most common phenotypes, each of which is worth more than 2000 in patients. We apply the ICD-9 codes to be the prediction label (a widely-used, though incomplete, surrogate for the phenotype). The phenotype disease name, ICD-9 code, disease type, and the number and percentage of patients for each phenotype in MIMIC-III are reported in Table 4.3.

Table 4.3: Descriptive Statistics of Phenotypes

Phenotype	**ICD-9**	**Type**	**# Patients (%)**
Essential hypertension	4019	chronic	13399 (43.2)
Coronary atherosclerosis of native coronary artery	41401	chronic	8208 (26.5)
Atrial fibrillation	42731	mixed	7525 (24.3)
Congestive heart failure	4280	mixed	6473 (20.9)
hyperlipidemia	2724	chronic	5387 (17.4)
Acute respiratory failure	51881	acute	4329 (14.0)
Pure hypercholesterolemia	2720	chronic	3874 (12.5)
Esophageal reflux	53081	chronic	3629 (11.7)
Pneumonia	486	mixed	2577 (8.3)
Chronic airway obstruction	496	chronic	2360 (7.6)

4.3.3.2 Implementation details

We describe the compromises made in order to feasibly train such a large model on GPUs, as well as the necessary trade-off in the experiments. Notably, the Hierarchical Transformer Networks require smaller BERT models than what are normally used, even when utilizing multiple GPU architectures. To achieve a fast and effective optimization, we implement an exponential decay with linear warmup for learning rate decay.

Distributed Training: As the models become larger and more complex, the computational resources and input texts are always limited to training those models on a single GPU or even TPU. Also, our method includes substantially longer sequence lengths (numerous thousands of words) than traditional GPU training can afford.

For example, the BERT model has a limited input of 512 word-pieced tokens. To overcome resource constraints and increase the input length, we apply mirrored distribution to distribute training over several GPUs. We train our proposed model on four NVIDIA Tesla V100 GPUs (32G), and the batch size is increased four times. On the other hand, each training takes about the same amount of time with either 1 GPU or $N+$ GPUs, resulting in an overall time reduction of four-fold assuming the training steps are the same.

4.3.4 Performance Comparison

4.3.4.1 Compared Baselines

The proposed Hierarchical Transformer Networks are compared with the following baselines:

BIGBIRD: Zaheer et al. (2020) extended the BERT model to longer sequences with sparse attention mechanisms, which is assumed to be the current state-of-the-art method for long-sequence text classification. BIGBIRD achieves good performances with more efficient architectures than traditional BERT, and it uses efficient attention to reduce the complexity while still preserving the model capacity. It can handle sequence lengths up to eight times longer than what was previously possible using similar hardware. In our case, we implement BIGBIRD for each document at the word-level and apply a fully-connected layer for the output probability. The BIGBIRD utilizes a flattened representation of texts directly from word to label, not considering hierarchical structure. Although it is not a hierarchical model, it can feed the same input length as the hierarchical models, so we consider this also

valuable to be implemented as one baseline.

HAN: The Hierarchical Attention Network (HAN) model is described in Section 4.2.2. The model at each level utilizes Bi-LSTMs and context-based attention mechanisms.

BERTLSTM: We additionally build a variant of the proposed model, denoted as BERTLSTM, in which the Transformer-based encoders are changed to Bi-LSTMs. We think this architecture also captures longitudinal and hierarchical information by using multi-level Bi-LSTMs. This enables us to evaluate the exact improvement in performance gained by the top-to-bottom Transformer architecture. This model is initially FTL-Trans (Zhang et al., 2020) updated to the patient level.

To ensure a fair comparison, we enable the hierarchical models to contain the same number of parameters (i.e., 5.6 million parameters in HAN, BERTLSTM, and the proposed model), while the BIGBIRD remains the same as in the released version (because the model is fixed). We carefully select the hyper-parameters to meet this comparison requirement.

4.3.4.2 Evaluation Metrics

We report predictive performances using the AUC, PRC, precision, recall, and F1-score. The use of PRC in addition to AUC tries to decrease variation caused by imbalanced class distributions since the Precision-Recall curve is well-suited for detecting infrequent situations. Patient cohorts are divided into training, development, and test sets in an 8:1:1 ratio. More precisely, at the completion of each epoch (a full run through the training set), we compute the loss on the development

Table 4.4: Overall Performance Comparisons

	Macro-AVG of 10-phenotype prediction				
	AUC	PRC	Precision	Recall	F1
BIGBIRD	0.7497	0.4647	0.6513	0.3515	0.4421
HAN	0.8845	0.6608	**0.7037**	0.5546	0.6033
BERTLSTM	0.8838	0.6483	0.6712	0.5733	0.5919
Our Model	**0.9096**	**0.7024**	0.7003	**0.6342**	**0.6462**
	In-hospital mortality prediction				
	AUC	PRC	Precision	Recall	F1
BIGBIRD	0.8769	0.8139	0.6924	0.7049	0.6986
HAN	0.9610	0.8992	0.7837	**0.8356**	0.8088
BERTLSTM	0.9608	0.8946	0.8740	0.7283	0.7945
Our Model	**0.9677**	**0.9032**	**0.8810**	0.7603	**0.8162**

set, and early stopping is activated when the loss continually increases for three consecutive epochs.

4.3.4.3 Results

Table 4.4 reports the predictive performance of in-hospital mortality and ten phenotypes (macro-averaged scores). For each phenotype, we also report the performance per-phenotype in PRCs as shown in Table 4.5. Notably, the Hierarchical Transformer Networks perform better than other alternatives for both in-hospital mortality and phenotypes in AUC, PRC, and F1-score.

We notice the performances of this flattened model, BIGBIRD, performs considerably worse than the other three hierarchical models in all tasks. So we think a more appropriate use case of BIGBIRD would be using it for efficient and effective

training in long-text document classification. In our case, we have a strong hierarchical structure due to the large number of notes in MIMIC, so the contributions from the hierarchical levels are important.

The performances of HAN and BERTLSTM are approximately the same. The advantages of Hierarchical Transformer Networks over BERTLSTM are significant in phenotype predictions, with improvements of 0.0258 in AUC, 0.0541 in PRC, and 0.0542 in F1-score. And Hierarchical Transformer Networks have relatively small improvements of 0.0251 in AUC, 0.0416 in PRC, and 0.0429 in F1-score, compared to HAN. This indicates the Transformers at all levels make a consistent contribution to the performance improvement. The application of BERT models at the word level has such a significant effect on prediction. Note that we only apply one layer of encoder in our proposed model, which already yields the best performance across alternatives. According to findings from the Ablation Study Section 4.3.5, the model still has a room to improve by increasing the model size and incorporating more data. Thus, we believe that the great potential of the Hierarchical Transformer Networks would outperform the existing advanced state-of-the-art methods in clinical outcome predictions.

Additionally, we find that Hierarchical Transformer Networks generate the best PRCs for both in-hospital mortality and almost all phenotypes. Considering that PRC is an important metric in machine learning prediction, accurately identifying the positive samples is necessary. This is also important for predicting clinical outcomes. A better PRC suggests that the Hierarchical Transformer Network is more accurate in detecting positive examples without mistakenly classifying nega-

Table 4.5: PRC scores of Different Models for All Phenotypes

ICD-9	BIGBIRD	HAN	BERTLSTM	Our Model
4019	0.7590	0.7817	0.8148	**0.8166**
41401	0.6967	0.9131	0.8938	**0.9163**
42731	0.6589	0.8771	0.8963	**0.8995**
4280	0.5734	0.7592	**0.7675**	0.7665
2724	0.4985	0.6940	0.7309	**0.7384**
51881	0.4068	0.6277	0.6051	**0.6396**
2720	0.4064	0.4522	0.2650	**0.5594**
53081	0.4073	0.6259	0.6532	**0.6754**
486	0.1228	**0.4131**	0.3587	0.4084
496	0.1167	0.4640	0.4976	**0.6037**
Macro_AVG	0.4647	0.6608	0.6483	**0.7024**

tive examples as positive, which is a more desirable method, particularly for clinical phenotype prediction.

4.3.5 Ablation Study

A significant portion of the performance of the transformers is dependent on the proper configuration of hyper-parameters. We explore numerous important factors that impact predictive accuracy, robustness, and efficiency in order to determine the best trade-off. This is fundamental for the proposed model because it necessitates the proper selection of hyper-parameters to make the model manageable in size.

4.3.5.1 Input Text Lengths

The released BERT models have a fixed sequence length of 128, which is significantly longer than the sentences of clinical notes. As presented in Table 4.3, the

Table 4.6: Performance of Hypertension with Different Input Lengths

Sequence length at each level [Percentile]			**Hypertension**	
Patient	Document	Sentence	AUC	PRC
22 [80th]	50 [80th]	64 [96.7th]	0.8722	0.8327
34 [90th]			0.8720	0.8337
16 [70th]			$\downarrow$ 0.8623	$\downarrow$ 0.8183
	85 [90th]		0.8733	0.8299
	37 [70th]		$\downarrow$ 0.8655	$\downarrow$ 0.8209
		128 [98.6th]	0.8744	0.8309
		32 [90th]	$\downarrow$ 0.8546	$\downarrow$ 0.8147
		22 [80th]	$\downarrow\downarrow$ 0.8347	$\downarrow\downarrow$ 0.7997

average number of word-pieced tokens per sentence is approximately 19. So it may be unnecessary to pad towards 128 tokens at the word level. However, removing an excessive number of tokens might have a negative impact on the capability of the pre-trained model. Hence, evaluating such a trade-off would be useful. We test the effectiveness of hypertension phenotype prediction at all three hierarchies using input sequences of varying lengths. The performances of different settings are reported in Table 4.6. We denote the first non-header row as the **base** input, where the models contain 80th percentile data length at the patient and document level, and 64 word pieces at the sentence level.

We first examine the results of different sequence lengths at the sentence level, or the number of tokens in a sentence, shown in the last row in Table 4.6. Even though the sequence length with 128 tokens has reached the 98.6th percentile, the

performance does not sizably improve (i.e., from 64 to 128, the AUC slightly increases by 0.0022). However, performance begins to decline gradually from an input size of 32. For lengths of 32 and 22, they do not perform well (with AUCs of 0.85 and 0.83) even though the input size is already up to the 90^{th} and 80^{th} percentiles, respectively. Thus, we believe that removing a substantial number of tokens from the initial input size of 128 does actually decrease the competence of the pre-trained model.

The results with sequence lengths at the patient and document levels (i.e., the number of notes and sentences) are shown in the Patient and Document columns. We experiment with 90^{th}, 80^{th}, and 70^{th} percentile data. All three settings yield an approximately comparable performance, with AUC scores of around 0.86 to 0.87. It is reasonable to have poor performance with 70^{th} percentile data (0.86+), but it makes a rather minor difference between 80^{th} and 90^{th} percentiles (0.87+).

4.3.5.2 BERT Variations

We ebvalute distilled BERT models with smaller sizes at the word level, including BERT_{tiny}, BERT_{mini}, BERT_{small}, BERT_{medium}, BERT_{base} (Turc et al., 2019). Given the same memory limits, we feed into the maximum sequence length for each distilled model, and we investigate if larger models would generate higher performances even with smaller sequence lengths.

Each BERT model is evaluated with three different settings: 1. The maximum length that the memory can afford (*Max Sequence Length*); 2. As BERT_{base} incorporates only 6 documents, all the other models are fed with the same 6 documents

Table 4.7: Performance of Hypertension with Distilled BERT Models

MODELS	Settings	Hypertension	
	Max Sequence Length	AUC	PRC
BERT$_{tiny}$	D50_S75_W128	0.8750	0.8181
BERT$_{mini}$	D40_S60_W64	0.8706	0.8066
BERT$_{small}$	D25_S50_W64	0.8863	0.8333
BERT$_{medium}$	D12_S50_W64	**0.8869**	**0.8365**
BERT$_{base}$	D6_S50_W64	0.8788	0.8178
	Last Six Notes		
BERT$_{tiny}$	D6_S50_W64	0.8660	0.8115
BERT$_{mini}$		0.8776	0.8213
BERT$_{small}$		0.8645	0.8040
BERT$_{medium}$		0.8763	**0.8231**
BERT$_{base}$		**0.8788**	0.8178
	Discharge Summary		
BERT$_{tiny}$	D1_S50_W64	0.8497	0.8030
BERT$_{mini}$		0.8496	0.7978
BERT$_{small}$		0.8627	0.8094
BERT$_{medium}$		0.8503	0.8036
BERT$_{base}$		**0.8649**	**0.8161**

(*Last Six Notes*); 3. Only discharge summary is fed into the model (*Discharge Summary*). All other hyper-parameters are the same across all BERT models. Only the BERT models applied at the word level and the input sequence lengths are different.

As shown in the column *Max Sequence Length* of Table 4.7, different models have different max input lengths (max_seq_len:D_S_W) that can be incorporated into 4 the GPU memories (128G) at their maximum capacity.

Notably, the max document length for BERT$_{medium}$ is only 12, but the perfor-

mance of BERT$_{medium}$ achieves the best AUC (0.8869) and PRC (0.8365) among all other combinations. For BERT$_{tiny}$, BERT$_{mini}$, and BERT$_{small}$, even though these three models incorporate many more documents than BERT$_{medium}$, the performances of them are still slightly worse than BERT$_{medium}$. Interestingly, BERT$_{base}$ performs worse than BERT$_{small}$ and BERT$_{medium}$.

Meanwhile, we investigate the impact of keeping the document length fixed at the BERT$_{base}$ max capacity of 6 documents. We run all other distilled models on the same 6 documents to understand if larger models would perform better than smaller models with the same input data. As presented in the column *Last Six Notes* of Table 4.7, we notice that BERT$_{base}$ achieves the best AUC and BERT$_{medium}$ achieves the best PRC.

Furthermore, we evaluate our model capacity by using only one document to predict the phenotype. We only process the discharge summary to predict whether the patient has hypertension. This would be more challenging than using all the notes because we only have a small portion of the data. We want to see if the proposed hierarchical architecture can still be used with the same architecture and achieve good performance. As reported in the *Discharge Summary* column of Table 4.7, the models continue to perform reasonably well, with an AUC of around 0.85. The best AUC (0.8649) and PRC (0.8161) were achieved by BERT$_{base}$.

However, compared to the performances that extensively use the majority of notes to make predictions, the results using only one note are worse. For all BERT models, the performances with the max sequence length and the last six notes outperform those only using the discharge summary. Thus, we show the necessity of

incorporating as many documents as possible. This is more important when the phenotype is such that it is hard to get a satisfactory performance. Adopting all possible notes into the model would yield sufficient room for improvement. Given the results of the above experiments, along with the general mantra "more data and larger models", we conclude that sufficient data is more crucial and would further improve the performance even if the model size may not be the largest. We therefore provide an applicable recommendation for those cases with less GPU memory: we should first ensure that we contain sufficient data, then select the larger model.

4.3.5.3 Transformer Encoder Variations

We evaluate different settings in the sentence- and document-level transformers, and the results are shown in Table 4.8. Unless specified, other hyper-parameters identical to best-performing model.

Numbers of Encoder Layers: We experiment with various encoder layers ($L = 1, 2, 4, 6, 8$). Table 4.8(A) shows that the model with 2 encoder layers achieves the best AUC (0.8722) and PRC (0.8327).

Notably, models with fewer layers ($L = 1, 2$) generally perform better than those with more layers ($L = 4, 6$).

Although this is opposed to the general mantra that larger models yield better performance, we assume it is because extreme model sizes might lead to an improvement bottleneck if the model is only used for fine-tuning classification.

Pooling Strategy: We also compare different pooling strategies on how to aggre-

Table 4.8: Performance of Hypertension with Transformer Variations

	Variations	**Hypertension**	
	Number of Layers	AUC	PRC
(A)	1	0.8674	0.8218
	2	**0.8722**	**0.8327**
	4	0.8645	0.8199
	6	0.8672	0.8213
	8	0.8684	0.8285
	Pooling Methods		
(B)	first	0.8683	0.8214
	mean	0.8702	0.8295
	max	0.8675	0.8222
	mean_max	**0.8722**	**0.8327**
(C)	w/o	0.8700	0.8294
	Positional Encoding	**0.8722**	**0.8327**
(D)	w/o	0.8558	0.7887
	Adaptive Segment	**0.8722**	**0.8327**

gate the representations from the previous level to the next. Table 4.8(B) finds that mean_max pooling is generally the best-performing pooling method.

Positional Encoding: Excluding positional encodings has a slight negative impact on performance, as shown in Table 4.8 (C). Thus, position-sensitive information is necessary for each representation unit to incorporate the orders of words, sentences, and documents.

Adaptive Segmentation: The results in Table 4.8 (D) show that there are significant decreases in AUC and PRC if we remove the adaptive segmentation. If clinical notes for the same patient are all independent without proper segmentation, the

effect is clearly reflected in the performance (i.e., only 0.8558 in AUC and 0.7887 in PRC).

CHAPTER 5

PATIENT-ORIENTED REPRESENTATION

In this chapter, we will introduce how we apply the methods developed in the previous chapters to generate **patient-oriented language representation**. We all know that the capabilities of clinical NLP methods have improved significantly in recent years, but clinical texts are still largely underused in patient-oriented clinical research and patient care. A key reason behind this is that NLP methods for clinical problems are mostly proposed and varied with certain tasks, for instance, the detection of specific diseases in free texts. But they generally can not satisfy the requirements of patient-oriented clinical research. Such a gap between the competence of clinical NLP methods and the requirements of patient-oriented clinical problems largely hinders the potential of NLP methods to tackle patient-oriented problems.

One way to mitigate this gap is to combine deep representation learning with large-scale textual data to generate patient representations. More specifically, we will build mathematical representations from EHR patient clinical notes covering different aspects of patients' health conditions. With the power of deep representation learning, we hope to make extensive use of clinical notes to learn semantic representations of patients and consider the complexities of patients from a comprehensive and holistic view.

We broaden this patient-oriented notion with clinical outcome-targeted super-

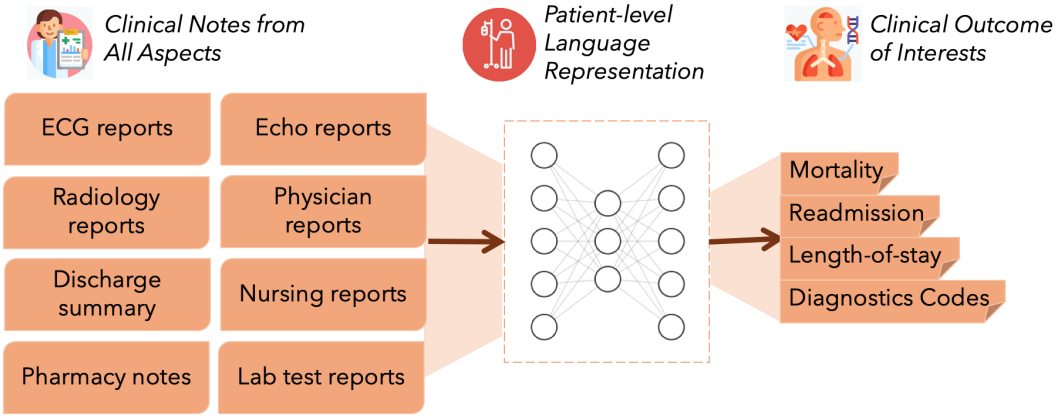

Figure 5.1: Clinical Outcome-targeted Supervised Learning

vised learning (Figure 5.1). Some clinical outcomes are important for patient-oriented clinical research, and they can be easily extracted from structured tables. For example, mortality, readmission, length-of-stay, and diagnostic ICD-9 codes are standard clinical outcomes of interest that are important to support clinical decisions. If we can make effective use of this data, we will be able to cover a wide range of resources for patient-oriented clinical outcomes while reducing the need for human annotation. One viable approach is to use structured EHR data as a source of supervision and develop supervised models that map clinical notes to clinical outcomes.The way it works is that the supervised models can connect clinical notes to clinical outcomes end-to-end directly. Because clinical outcomes are essential clinical knowledge about the patient, the model can therefore deeply encode notes with clinically-meaningful information about the patient.

In the following two sections, we introduce two advanced learning techniques, including multi-task learning and transfer learning, to assist in the development of patient-oriented language representation.

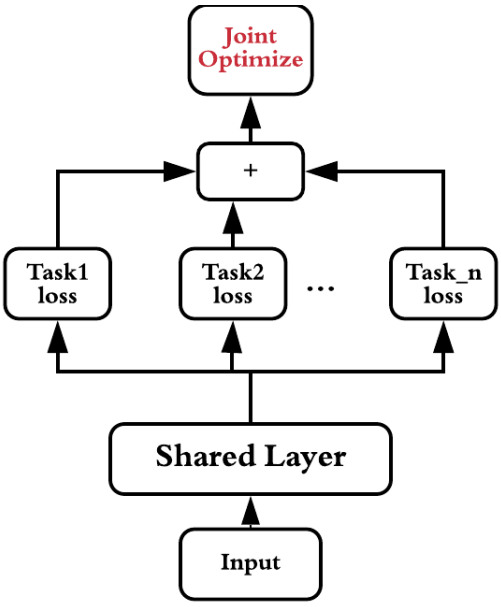

Figure 5.2: Multi-task Learning Joint Optimization

5.1 Multi-task Learning

5.1.1 Backgrounds

Multi-task learning, or MTL, is a type of machine learning where related tasks are selected to be trained concurrently and the parameters are shared in part (Figure 5.2). When compared to training a single task, MTL increases the model's capacity for generalization and performance by leveraging information across tasks. MTL is effective because of its ability to help the task engage, and it functions as a regularization technique in machine learning by adding an inductive bias that is beneficial to the training.

Numerous attempts have been made to process patient data from clinical notes to develop meaningful representations. However, these attempts are generally fo-

Figure 5.2 retrieved from Si and Roberts (2019).

cused on a single prediction task and overlook the possibility of shared knowledge across related tasks. However, superior performance on one task does not always imply superior performance on other tasks; information sharing among tasks may result in higher performance than the individual task. Despite this advantage, there has been very little research evaluating the feasibility and contribution of training multiple tasks to clinical outcome predictions. We hypothesize that MTL models have the capacity to establish more generic representations, which would be effective for tasks other than those for which they are trained.

5.1.2 Related Work

With the development of deep learning, studies examining the application of MTL to enhance training efficiency and predictive accuracy have continually been proposed. Based on the survey by Zhang and Yang (2021), we understand the theoretical basis of MTL and its potential directions. Ruder (2017) provided an in-depth study of MTL using deep learning models and found that deep learning significantly accelerates computation and increases the possibility of achieving MTL. Many machine learning studies have successfully implemented MTL on open-domain tasks such as image recognition (Li et al., 2014; Elhoseiny et al., 2016; Chowdhuri et al., 2019; Cao et al., 2018; Pasunuru and Bansal, 2017), NLP (Collobert and Weston, 2008; Dong et al., 2015; Rei, 2017; Liu et al., 2017a; Crichton et al., 2017; McCann et al., 2018), and speech recognition (Toshniwal et al., 2017). It is already well known that MTL can improve training efficiency when the model is capable of sharing knowledge between related tasks (Kumar and Daumé III, 2012). Even

when encountered with issues of negative effects, MTL attempts to deliver insights about the tasks, and it has been primarily applied in biomedical discoveries, including biological function (Yang et al., 2009; Fa et al., 2018) and drug discovery (Ramsundar et al., 2015).

For now, clinical events are being used as an instance of multi-task EHR learning where certain parameters of the model are shared while others are specialized, as it is a difficult problem in medicine because of the intricacy of the surrounding situations and the inconsistent data from a wide variety of diverse perspectives(Caruana et al., 1995). Futoma et al. (2017) developed a Multi-task Gaussian Process (MGP) RNN model to predict sepsis at an early stage using physiological data (i.e., vital signs and lab variables). Harutyunyan et al. (2019) integrated time-series variables into MTL networks to predict four clinical benchmarks, which perform better compared to solely feature engineering. Ngufor et al. (2015) investigated ways of clustering related tasks in MTL to facilitate knowledge transfer and predict blood transfusion (RBC) procedure outcomes. Razavian et al. (2016) evaluated three MTL neural networks with two CNN variations and a LSTM variant, against single task learning (STL) baselines for forecasting the disease onset entirely with longitudinal lab variables. Wiens et al. (2016) adjusted MTL to build risk stratification models with time-series features, and different groups of patient cohorts are considered as related tasks. Wang et al. (2014b) introduced a MTL approach for disease onset predictions with ICD-9 codes. Nori et al. (2017) showed that MTL models outperformed STL models in forecasting the mortality of ICU patients using demographic data and diagnostic codes. Lopez-Martinez and Picard (2017) used physiological

variables including skin conductance and ECG readings with MTL to predict pain recognition. In general, these studies demonstrate the increasing significance of implementing MTL in clinical settings with clinical EHR data such as laboratory testing or physiological values. However, despite the fact that a few studies have utilized MTL for modeling clinical events, earlier research has shown conflicting findings (Ding et al., 2018). Also, there has been insufficient work about the implementation of MTL to learn from clinical notes. To our knowledge, we are one of the earliest teams that first integrated MTL to learn patient representation from unstructured clinical notes (Si and Roberts, 2019).

5.2 Transfer Learning

5.2.1 Backgrounds

Another powerful idea in deep learning that can be used to generate transferable patient-level representation is known as transfer learning. With this, we can take the knowledge the neural networks have learned from one task and apply that knowledge to a separate task. For typical machine learning or deep learning models, we usually collect datasets, randomize a model from scratch, and train the model on this dataset. If we have another task, we will do this whole process again. The same with a third task, a fourth task, and so on. For humans, we do not learn like that. Instead, we use all the knowledge we have already had to solve the new problem. So transfer learning is one way to try to do that in machine learning.

The main method of transfer learning for NLP has two step (Figure 5.3). The first step is called the pre-training step, which is actually computationally inten-

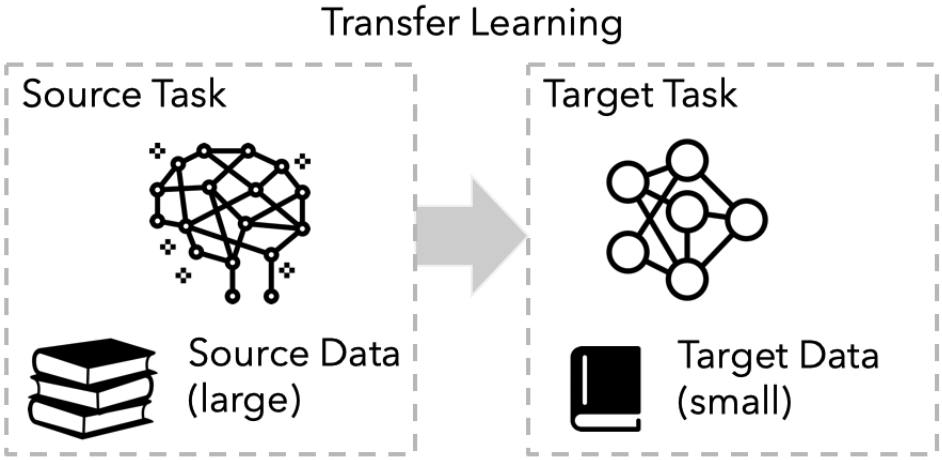

Figure 5.3: Transfer Learning in NLP

sive, and during this step, we collect as much data as we can and train a model on all of the data, which ends up with a general-purpose representation or a general-purpose model. The model is a set of pre-trained weights, so in the second step, which we call fine-tuning, we start from this pre-trained model and now we adapt it to the exact task. The new task normally has a small sample size. The reason transfer learning can be helpful is that a lot of the low-level patterns, such as the structure and nature of what language looks like, are already known, and some of that knowledge will be useful. So, having learned to understand language from the source task, it may have learned enough to help your new task learn a bit faster or learn with less data.

5.2.2 Related Work

Studies with transfer learning to facilitate EHR patient modeling are relatively innovative at this moment. DoctorAI was the first and most notable effort to use deep learning models to learn patient representations from large datasets in EHRs, where

Choi et al. (2016a) used recurrent neural networks to learn dense patient vectors from sequential clinical events. This research also established the benefit of transfer learning from large-scale clinical problems to downstream tasks with limited data. Besides the supervised learning approach, DeepPatient (Miotto et al., 2016) developed an unsupervised model based on a stacked denoising autoencoder (SDA) for learning patient representations from a longitudinal EHR data warehouse. Following the same idea as DeepPatient, Sushil et al. (2018) investigated unsupervised approaches for learning patient representations from unstructured notes and fed them as input embeddings into clinical predictions. Dligach et al. (2019) evaluated transfer learning by pre-training a CUI-driven CNN model with clinical notes from MIMIC-III and then applying the CUI-driven model by extracting vectors to predict phenotypes on the i2b2 2008 challenge (Uzuner, 2009). Kemp et al. (2019) conducted a more current study in which they established hierarchical modeling of clinical notes to develop patient representations and also illustrated the impact of pre-training. In the end, Steinberg et al. (2021) suggested that language models are effective for developing competent patient representations. Their research has shown that leveraging the word2vec model to patient representation can improve clinical prediction performances as well as knowledge transfer from the full cohort to certain tasks. In sum, these investigations validate the premise that developing a solid patient representation is conceivable using resources directly from unstructured notes and also demonstrate the potential for medical language transfer learning.

However, in previous work on EHR transfer learning, a feature-based method

was extensively used, that is, to extract patient vectors from the intermediate or the last layers of neural networks and to consider them as the input to the fine-tuning task. Because this extraction approach is task-specific, performance gains on other tasks may be marginal. Motivated by the notion of pre-training and fine-tuning from NLP, we propose a model-wise transfer learning approach to train patient representations from free-text medical language. More importantly, we are one of the first efforts to use state-of-the-art transfer learning to develop transferable and generic patient representations from clinical notes. In the next section, we will introduce the study that builds such patient representations and evaluates them on low-prevalence phenotypes.

5.3 Generalized and Transferable Patient Language Representation for Phenotyping with Limited Data

5.3.1 Introduction

The purpose of representation learning is to effectively develop a semantically meaningful representation from original data. This problem has received considerable attention in the field of natural language processing. Solid representations that convert raw data into meaningful information are important for machine learning models to perform effectively. With the concept of representation learning in mind, we propose to learn meaningful representations of EHR data, a process we refer to as patient representation learning. The motivation behind patient representation learning is to construct a mathematical model of a patient from original EHR data, which in turn is sparse and high-dimensional in itself. The majority of

techniques for patient representation learning are task-customized, which generate the representation only for certain predictions at each time. An ideal patient representation, on the other hand, would be robust and widely applicable to a wide spectrum of prediction tasks. A good representation should be especially beneficial for "small data" problems when there are insufficient instances to develop good representations using task-specific models.

As discussed in the previous subsection, extensive studies in computer vision and natural language processing have shown the effectiveness of transfer learning, in which a practical technique is to fine-tune a downstream task using large-scale pre-trained models, such as ImageNet in the computer vision domain Deng et al. (2009) and BERT in the NLP domain (Devlin et al., 2019). This formula would also be well suited for developing a transferable patient representation, given that pre-training with a large dataset would transfer knowledge of medicine to additional clinical tasks, hence improving prediction results. Pre-training begins with the development of a source task, and we hope that the framework (i.e., model and objectives) will incorporate a broader picture of the patient's information, much like how clinicians diagnose patients based on thorough knowledge of patients. Accordingly, generalized patient presentations can be achieved with pre-training. In order to accomplish this aim in the most robust way possible, multi-task joint learning among multiple and yet associated clinical predictions is intended for the task of composing the pre-training source tasks.

Our proposed representation framework integrates multi-task learning of phenotype predictions in source tasks and fine-tuning of target tasks. In more detail,

the model is pre-trained using a large corpus of clinical notes with supervised learning targets of numerous clinical outcomes, namely distinct but related phenotypes with high frequencies. We anticipate that this framework is capable of comprehending generalized and uniform representations because multiple related common phenotypes would cover the vast proportion of patients. We then apply the representations to be further fine-tuned on low-prevalence phenotype predictions, to verify the robustness of the pre-trained model, which is also our main motivation for classifying low-prevalence phenotypes with unstructured clinical notes. This task is often complicated to perform well due to the scarcity of positive samples. We expect that if the pre-trained model is further fine-tuned, the model will transfer knowledge about patients and hence assist with the low-prevalence phenotyping predictions. A generically-idealized patient representation should promote ”limited data” problems, in which only a small amount of data is available to get superior performance solely with its data. As such, we define the low-prevalence phenotype classifications as downstream tasks to be fine-tuned further starting from the pre-trained multi-task model and also test the generalizability of the results.

As hypothesized, our experiment findings indicate that when compared to single-task pre-training and no transfer learning, multi-task pre-training always increases performance of low-prevalence phenotyping predictions. Notably, we only get to conduct the pre-training once and then utilize this generic model for a variety of phenotypes. In terms of phenotyping algorithms, it is difficult for NLP-based phenotyping methods to execute the wide variety of phenotypes shown in this work. This is why we highlight the stability and robustness of the proposed method: it

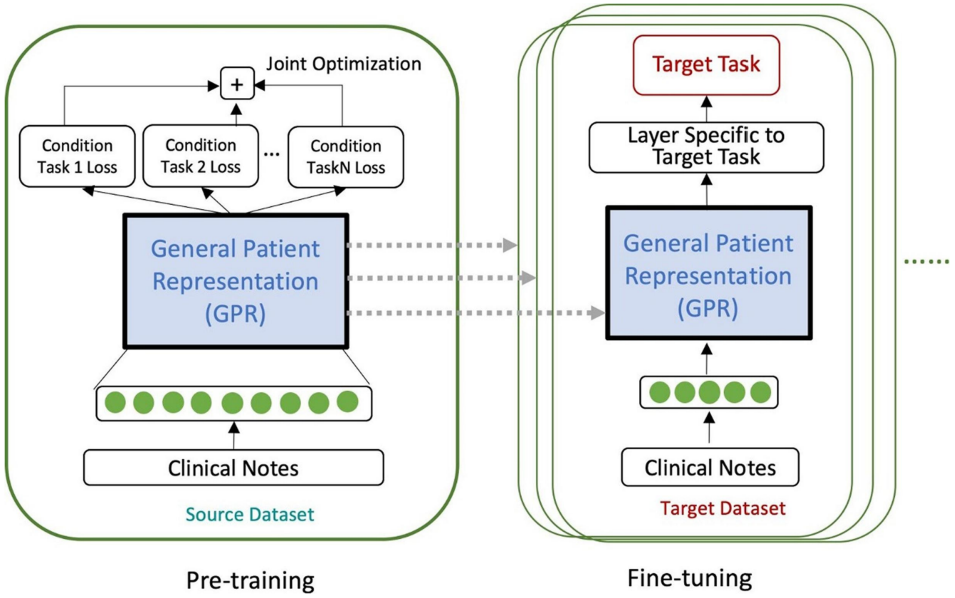

Figure 5.4: Overall Pre-training and Fine-tuning Architecture

consistently outperforms the baselines, but when it does not, the score differences are reasonably small.

5.3.2 Methods

5.3.2.1 Overall Framework

This section describes the overall implementation of the representation framework, shown in Figure 5.4. Essentially, the overall process is divided into two stages, including pre-training and fine-tuning. During pre-training, we use supervised multi-task learning, leveraging latent information across various tasks. The supervised classifier is optimized towards a variety of high-prevalence diseases in parallel. During fine-tuning, the downstream task is low-prevalence phenotyping, which is

Figure 5.4 retrieved from Si et al. (2021a).

one of the most challenging problems due to the dearth of data. We apply low-prevalence phenotyping to be continually fine-tuned. More particularly, we focus on phenotypes that are particularly unusual in MIMIC-III data, specifically those characterized by ICD-9 having fewer than 550 patients, which ends up with 78 low-prevalence phenotypes that match this criterion. Fine-tuning at this stage entails retrieving from source tasks and adapting the pre-trained models to the target tasks, as well as starting training with the pre-trained parameters.The model parameters are updated constantly depending on the target phenotype's labels. Each target also has its own distinct, fully-connected layer before the actual prediction.

The model in pre-training and fine-tuning for learning patient representations is a HAN model, described in Section 4.2. We chose this model because it would be capable of dealing with the hierarchical structures and temporal relations of clinical notes. Also, it is not that large as hierarchical transformer networks, which would lead to overfitting on low-prevalence phenotypes. Considering that we are assessing different transfer learning approaches instead of getting the state-of-the-art scores, we determine this model with a balance between efficiency and effectiveness. We refer to Section 4.2 for more details about this neural network. In terms of the model hyper-parameters, we report as follows: word embeddings of 50 dimensions, the hidden unit of LSTM: 100, the output size of attention mechanisms: 200. This yields a HAN model with a total trainable parameter of 653,101.

5.3.2.2 High-prevalence Phenotyping-guided Pre-training

To achieve a generic patient representation, we develop source tasks for pre-training that aim to be comprehensive by covering a wide range of patient data aspects.We develop a multi-task learning model on associated high-prevalence phenotypes with the goal of ensuring that patient information is captured and integrated into the pre-trained source tasks. We focus on high-prevalence phenotypes from three organ systems in particular, including circulatory, respiratory, and genitourinary. These three organ systems are the three most frequent systems by ICD-9 counts. We choose the five most common phenotypes in each organ system based on patient frequency and pre-train the model to jointly learn these five phenotypes. The information about phenotypes per organ system is shown in Table 5.1.

Table 5.1: Top Five High-prevalence Phenotypes in Three Organ Systems

Circulatory		**Respiratory**		**Genitourinary**	
Disease Name (ICD-9)	**# patients**	**Disease Name (ICD-9)**	**# patients**	**Disease Name (ICD-9)**	**# patients**
Essential hypertension (401.9)	20,703	Acute respiratory failure (518.81)	7,497	Acute kidney failure (584.9)	9,119
Congestive heart failure (428.0)	13,111	Pneumonia (486)	4,839	Urinary tract infection (599.0)	6,555
Atrial fibrillation (427.31)	12,891	Chronic airway obstruction (496)	4,431	Chronic kidney disease (585.9)	3,435
Coronary atherosclerosis of native coronary artery (414.01)	12,429	pleural effusion (511.9)	2,734	Acute kidney failure with lesion of tubular necrosis (584.5)	2,287
Hypertensive chronic kidney disease (403.90)	3,421	Asthma (493.90)	2,195	End stage renal disease (585.6)	1,926

More specifically, in pre-training, we train a HAN model with MIMIC-III clinical notes. The word embeddings of 50 dimensions are fed into the input layer, and the model is a three-level HAN architecture that proceeds from words towards sentences, documents, and eventually to the patient. Pre-training objectives are to classify patients if they have at least one of the five high-prevalence phenotypes. To be more precise, we pre-train the model for each organ system with joint training of the top five phenotypes. The overall loss from all five tasks is minimized. For prediction labels, we utilize the ICD-9 code from structured EHR data as a surrogate for the phenotype. The labels for pre-training in multi-task learning consist of all five phenotypes. Namely, if the patient has ICD-9 codes for phenotypes A and B but without ICD-9 codes for phenotypes C, D, and E, then the label would be as follows: phenotype A as positive, phenotype B as positive, phenotype C as negative, phenotype D as negative, and phenotype E as negative. Because the pre-training is conducted within each organ system, we may refer to the five-phenotype pre-trained models as organ system-customized models. In the end, we achieve three pre-trained models, including the circulatory model, the respiratory model, and the genitourinary model.

5.3.2.3 Fine-tuning on Low-prevalence Conditions

In fine-tuning, we directly apply the pre-trained model to the target task and train the model from start to finish.The pre-trained model is provided with task-specific inputs, including word embeddings from the target corpus and labels of low-prevalence phenotypes extracted from structured data. The target task is initiated using param-

eters of the pre-trained model. At the output layer, a task-specific fully-connected network is added for final prediction, which is identical to the fine-tuning in BERT that converts the output state to a logit function for prediction probability.

As mentioned before, the primary goal of this research is to apply this pre-training and fine-tuning strategy to enhance prediction for relatively uncommon phenotypes, which would benefit a lot from the effective transfer learning technique. We identify phenotypes in three organ systems that are relatively infrequent with only 50 to 550 individual patients. 78 phenotypes (i.e., 38 circulatory phenotypes, 23 respiratory phenotypes, and 17 genitourinary phenotypes) are included. To eliminate cherry-picking, we conduct experiments on all 78 phenotypes. The information about these 78 phenotypes is shown in Table 5.2, Table 5.3, and Table 5.4. Because of the significantly imbalanced positive samples for low-prevalence phenotypes, we scale the loss functions with assigned coefficients in order to favor positive samples over negatives. The following formula is used to determine the weight assigned to each phenotype:

$$\text{weight-positive} = \frac{1}{\text{positive}} \times \frac{\text{total}}{2} \tag{5.1}$$

where *positive* represents the patient counts in positive samples, and *total* represents the entire patient cohorts (in this experiment we have 31,360 patients from MIMIC-III data).

Table 5.2: Low-prevalence Phenotypes in the Circulatory System

Disease name	ICD-9	# cases (Weight)	STL-related	Disease name	ICD-9	# cases (Weight)	STL-related
Acute systolic heart failure	428.21	492 (32)	428.0	Iatrogenic hypotension	458.2	233 (72)	428.0
Coronary atherosclerosis of autologous vein bypass graft	414.02	474 (33)	414.01	Cerebral atherosclerosis	437.0	196 (80)	401.9
Other late effects of cerebrovascular disease	438.89	465 (34)	401.9	Hypertrophic cardiomyopathy	425.1	183 (86)	401.9
Benign essential hypertension	401.1	454 (35)	401.9	Chronic combined systolic and diastolic heart failure	428.42	179 (88)	428.0
Late effects of cerebrovascular disease , hemiplegia affecting unspecified side	438.20	437 (36)	403.90	Malignant essential hypertension	401.0	172 (91)	401.9
Acute diastolic heart failure	428.31	432 (36)	401.9	Paroxysmal supraventricular tachycardia	427.0	161 (97)	427.31
Systolic heart failure, unspecified	428.20	416 (38)	428.0	Acute myocardial infarction of anterolateral wall, initial episode of care	410.01	142 (110)	414.01
Subdural hemorrhage	432.1	392 (40)	401.9	Hypertensive chronic kidney disease, benign, with chronic kidney disease stage I through stage IV, or unspecified	403.10	122 (129)	403.90
Sinoatrial node dysfunction	427.81	389 (40)	427.31	Combined systolic and diastolic heart failure, unspecified	428.40	110 (143)	428.0
Acute myocardial infarction of unspecified site, initial episode of care	410.91	354 (44)	414.01	Unspecified transient cerebral ischemia	435.9	96 (163)	401.9
Atherosclerosis of native arteries of the extremities with gangrene	440.24	327 (48)	427.31	Atherosclerosis of native arteries of the extremities with rest pain	440.22	88 (178)	414.01
Acute on chronic combined systolic and diastolic heart failure	428.43	327 (48)	428.0	Abdominal aneurysm, ruptured	441.3	76 (206)	401.9
Chronic total occlusion of coronary artery	414.2	292 (54)	414.01	Acute combined systolic and diastolic heart failure	428.41	73 (215)	428.0
Atherosclerosis of aorta	440.0	283 (55)	414.01	Unspecified late effects of cerebrovascular disease	438.9	69 (227)	401.9
Sub-endocardial infarction, subsequent episode of care	410.72	279 (56)	414.01	Unspecified cerebrovascular disease	437.9	64 (245)	401.9
Atherosclerosis of native arteries of the extremities with ulceration	440.23	264 (59)	414.01	Atherosclerosis of other specified arteries	440.8	63 (249)	414.01
Atherosclerosis of native arteries of the extremities with intermittent claudication	440.21	257 (61)	414.01	Cerebral thrombosis with cerebral infarction	434.01	60 (261)	401.9
Late effects of cerebrovascular disease, aphasia	438.11	240 (65)	427.31	Secondary cardiomyopathy, unspecified	425.9	53 (296)	428.0
Atherosclerosis of renal artery	440.1	233 (67)	414.01	Acute myocardial infarction of unspecified site, subsequent episode of care	410.92	53 (296)	414.01

Table 5.3: Low-prevalence Phenotypes in the Respiratory System

Disease name	**ICD-9**	**# cases (Weight)**	**STL-related**	**Disease name**	**ICD-9**	**# cases (Weight)**	**STL-related**
Post-inflammatory pulmonary fibrosis	515	544 (29)	486	Pulmonary congestion and hypostasis	514	155 (101)	511.9
Pneumonia due to Pseudomonas	482.1	430 (36)	486	Sinusitis (chronic)	473.9	149 (105)	493.90
Chronic respiratory failure	518.83	331 (47)	518.81	Edema of larynx	478.6	145 (108)	518.81
Acute edema of lung, unspecified	518.4	305 (51)	511.9	Malignant pleural effusion	511.81	132 (119)	511.9
Chronic obstructive asthma with (acute) exacerbation	493.22	299 (52)	496	Acute bronchitis	466.0	126 (124)	518.81
Pneumonia due to other gram-negative bacteria	482.83	264 (59)	486	Asbestosis	501	116 (135)	496
Bacterial pneumonia, unspecified	482.9	227 (69)	486	Acute upper respiratory infections	465.9	96 (163)	493.90
Pneumonia due to Klebsiella pneumoniae	482.0	226 (69)	486	Abscess of lung	513.0	86 (182)	486
Pneumococcal pneumonia [Streptococcus pneumoniae pneumonia]	481	194 (81)	486	Unilateral paralysis of vocal cords or larynx, partial	478.31	74 (212)	518.81
Bronchiectasis without acute exacerbation	494.0	191 (82)	486	Empyema with fistula	510.0	72 (218)	511.9
Empyema without mention of fistula	510.9	190 (83)	511.9	Stenosis of larynx	478.74	61 (257)	518.81
Methicillin resistant pneumonia due to Staphylococcus aureus	482.42	162 (97)	518.81				

Table 5.4: Low-prevalence Phenotypes in the Genitourinary System

Disease name	**ICD-9**	**# cases**	**Weight**	**STL-related**
Hematuria	599.7	509	31	599.0
Hydronephrosis	591	413	38	584.9
Chronic kidney disease, Stage IV (severe)	585.4	334	47	585.6
Hypertrophy (benign) of prostate with urinary obstruction and other lower urinary tract symptoms (LUTS)	600.01	314	50	599.0
Neurogenic bladder NOS	596.54	225	70	599.0
Hematuria, unspecified	599.70	216	73	599.0
Calculus of kidney	592.0	206	76	599.0
Gross hematuria	599.71	181	87	599.0
Secondary hyperparathyroidism (of renal origin)	588.81	169	93	585.9
Acute pyelonephritis without lesion of renal medullary necrosis	590.10	132	119	599.0
Calculus of ureter	592.1	111	141	599.0
Cyst of kidney, acquired	593.2	107	147	585.9
Hypertrophy (benign) of prostate	600.0	92	170	599.0
Pyelonephritis, unspecified	590.80	84	187	599.0
Nephritis and nephropathy, not specified as acute or chronic, with unspecified pathological lesion in kidney	583.9	84	187	584.9
Vascular disorders of kidney	593.81	83	189	584.5
Chronic glomerulonephritis in diseases classified elsewhere	582.81	67	234	585.9

5.3.3 Experiments

5.3.3.1 Baseline Methods

Our proposed method, multi-task transfer learning (MTL) is compared with three baseline methods of two types: single-task transfer learning (STL), and no transfer learning (No-transfer). The details of baselines are described in the following:

STL-highest: Single-task transfer learning of the high-prevalence phenotype with the highest patient count. Only one among five high-prevalence phenotypes with the largest number of patients is considered in the source task of pre-training. In other words, the pre-training task for this baseline in the circulatory, respiratory, and genitourinary organ systems is *Unspecified essential hypertension* (ICD-9: 401.9), *Acute respiratory failure* (ICD-9: 518.81), and *Acute kidney failure* (ICD-9: 584.9), respectively.

STL-related: Single-task transfer learning of the phenotype that is medically relevant to the target phenotype. The pre-training task includes only one of the five most medically relevant high-prevalence phenotypes for the downstream task. A practicing physician in internal medicine chose the source phenotype for each target task based on clinical knowledge, and the selection process was finished before training. The columns of *STL-related* in Table 5.2, Table 5.3, and Table 5.4 present the target task's most related high-prevalence phenotype.

No-transfer: This is a typical technique in machine learning, and the model is trained only with information from the target task per low-prevalence phenotype.

In the following result sections, we refer to **MTL**, **STL-highest**, **STL-related**, and **No-transfer**.

5.3.3.2 Data and Implementation Details

This work uses large-scale MIMIC-III clinical notes (i.e., nearly 2 million unstructured notes). Each note is tokenized using regular expressions and sentences are split with spaCy. We use ICD-9 codes to predict phenotypes. If a patient has an ICD-9 code, they are considered a positive example of that given phenotype. Other patients without the ICD-9 code are negative. The word embeddings are used as input features for both pre-training and fine-tuning.

In terms of prediction labels, the predictions are entirely patient-level. The *MTL* labels are the five most prevalent phenotypes in each organ system. The *STL-highest* labels are the phenotype with the largest patient count for each organ system. The *STL-related* labels are extracted from Table 5.2, Table 5.3, and Table 5.4. The *No-transfer* does not require pre-training. It only has the label if the patient is diagnosed with each low-prevalence phenotype, which is also the label in fine-tuning across all methods.

5.3.3.3 Evaluation Metrics

To prevent any data leakage, we reserve the test set and only utilize it to report the performance. That is to say, the pre-training and fine-tuning have no chance of learning from test data. We use a sigmoid loss function with the positive weights shown in Table 5.2, Table 5.3, and Table 5.4 to predict low-prevalence phenotypes. We report the AUC score for the performance metric. All four methods and all 78 phenotypes are experimented with, resulting in 312 AUC scores. We notice that limited test sets for particular phenotypes result in greater variance in the values.

We hope using AUC instead of F1 and accuracy sort of minimizes the variance since AUC is a reliable measurement.

To compare four different methods, we calculate and report the experiment findings as follows:

1. AUC values for all 78 phenotypes (Table 5.6, Table 5.7, Table 5.8)
2. Performance distributions of four methods in three organ systems with boxplots (Figure 5.5)
3. The number of phenotypes for which *MTL* perform better than three baselines (Table 5.9)
4. The number of phenotypes for which each method performs the best (Table 5.10)
5. If the method performs poorly, the number of phenotypes that are still within 90% of the highest AUC score (Table 5.10)
6. The average mean squared error (Avg-MSE) for four methods across three organ systems (Table 5.11)

5.3.4 Results

5.3.4.1 Pre-training Results

In pre-training, we obtain three five-task pre-trained models containing five-task circulatory model, five-task respiratory model, and five-task genitourinary model. We also pre-train 15 *STL* models for each of the five phenotypes across three organ

Table 5.5: Performances of High-prevalence Phenotypes

Circulatory	**MTL**	**STL**	**Respiratory**	**MTL**	**STL**	**Genitourinary**	**MTL**	**STL**
Essential hypertension	0.8041	0.8041	Acute respiratory failure	0.9107	**0.9112**	Acute kidney failure	0.8469	**0.8519**
Congestive heart failure	**0.9183**	0.9145	Pneumonia	0.8542	**0.8603**	Urinary tract infection	0.7423	**0.7468**
Atrial fibrillation	**0.9408**	0.9336	Chronic airway obstruction	**0.8378**	0.8048	Chronic kidney disease	**0.8664**	0.8550
Coronary atherosclerosis of native coronary artery	**0.9517**	0.9503	Pleural effusion	0.8539	**0.8560**	Acute kidney failure with lesion of tubular necrosis	**0.9105**	0.8882
Hypertensive chronic kidney disease	**0.8768**	0.8731	Asthma	**0.8449**	0.5734	End stage renal	**0.9752**	0.9412

systems. *MTL* and *STL* performances (AUCs) for each phenotype are shown in Table 5.5.

The *MTL* column means that each high-prevalence phenotype was evaluated using the pre-trained 5-task model on the test set, while the *STL* column means that each high-prevalence phenotype was evaluated on its corresponding *STL* model. In other words, the scores in the *STL* column of each row are derived from a single *STL* model, while the five scores in the *MTL* column are jointly derived from only one *MTL* model. Even though only one model is pre-trained in the *MTL* scenario, because there are five phenotypes being optimized during training, there would be five scores that correspond to these five phenotypes. Also, while our primary focus is on fine-tuning low-prevalence phenotypes, we believe that pre-training performance is also worth investigating.We observe that *MTL* has very little negative impact on the performances of high-prevalence phenotypes and yields performances that are well-matched in almost all tasks. Surprisingly, there are major improvements in most phenotypes from *STL* to *MTL*, and the highest improvement can reach up to 0.2715 in AUC from *STL* to *MTL* (i.e., asthma).

5.3.4.2 The Effectiveness of Pre-training

We continue to assess the effectiveness of transfer learning on the target task of low-prevalence phenotypes. First, to get an overall understanding of performances, we plot the distribution of AUCs across four methods in three organ systems, as shown in Figure 5.5. The exact AUC scores are reported in Table 5.6, Table 5.7, and Table 5.8. The performance of each organ system and each method is distributed and shown in a box plot, which depicts the median, the lowest and highest values, as well as the first and third quantiles or quartiles. We see that among four methods in all organ systems, the *MTL* has the most compact range, while the *No-transfer* has the most unstable distribution. Also, the median of *MTL* is always higher when compared to the other three methods.

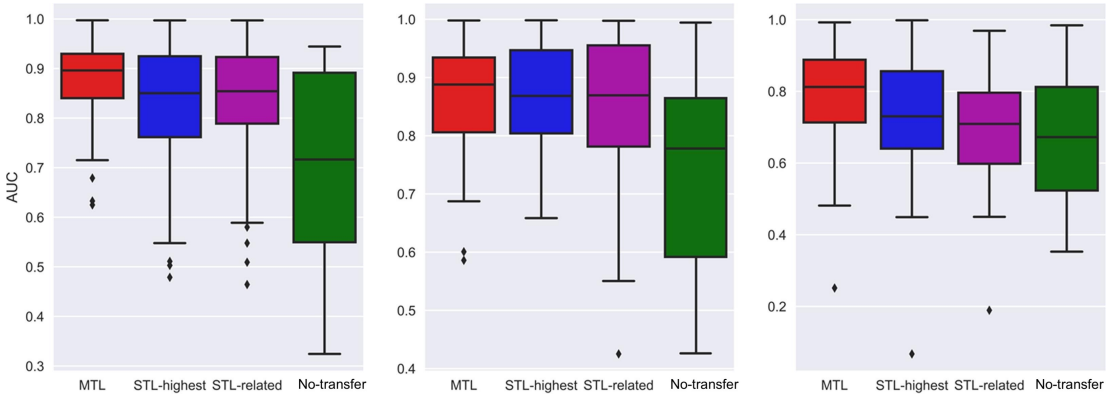

Figure 5.5: Box-plots of AUC Distributions

Figure 5.5 retrieved from Si et al. (2021a).

Table 5.6: Predictive Performances of the Circulatory System

Disease name	MTL	STL-highest	STL-related	No-Transfer
Acute systolic heart failure	0.9256	0.9238	0.9124	**0.9279**
Coronary atherosclerosis of autologous vein bypass graft	0.9627	**0.9662**	0.9403	0.9295
Other late effects of cerebrovascular disease	**0.8595**	0.8301		0.5741
Benign essential hypertension	0.8542	**0.8850**		0.6578
Late effects of cerebrovascular disease, hemiplegia affecting unspecified side	**0.9533**	0.9297	0.9057	0.4138
Acute diastolic heart failure	0.8801	0.8793		**0.9086**
Systolic heart failure, unspecified	0.8248	**0.8668**	0.5946	0.7846
Subdural hemorrhage	0.9036	**0.9640**		0.9063
Sinoatrial node dysfunction	0.931	0.9251	0.9359	0.9209
Acute myocardial infarction of unspecified site, initial episode of care	**0.8400**	0.8151	0.7982	0.7581
Atherosclerosis of native arteries of the extremities with gangrene	0.9448	0.9305	**0.9557**	0.9241
Acute on chronic combined systolic and diastolic heart failure	0.8448	0.8173	0.833	**0.8594**
Chronic total occlusion of coronary artery	0.9509	0.9405	**0.9532**	0.9444
Atherosclerosis of aorta	0.8158	0.8345	0.844	**0.8709**
Sub-endocardial infarction, subsequent episode of care	**0.9396**	0.8986	0.9043	0.4282
Atherosclerosis of native arteries of the extremities with ulceration	0.8312	0.8341	**0.9213**	0.6387
Atherosclerosis of native arteries of the extremities with intermittent claudication	0.8295	**0.9059**	0.8659	0.8893
Late effects of cerebrovascular disease, aphasia	**0.9590**	0.7527	0.6746	0.5184
Atherosclerosis of renal artery	**0.9014**	0.8544	0.7915	0.4861
Iatrogenic hypotension	**0.9223**	0.8896	0.8526	0.6864
Cerebral atherosclerosis	**0.7836**	0.7524		0.5232
Hypertrophic cardiomyopathy	**0.9951**	0.9378		0.6124
Chronic combined systolic and diastolic heart failure	**0.9043**	0.5110	0.8916	0.8922
Malignant essential hypertension	**0.8528**	0.8480		0.6048
Paroxysmal supraventricular tachycardia	**0.7152**	0.6640	0.5093	0.7146
Acute myocardial infarction of anterolateral wall, initial episode of care	0.9124	0.5029	**0.9240**	0.9172
Hypertensive chronic kidney disease, benign, with chronic kidney disease stage I through stage IV, or unspecified	**0.8408**	0.7920	0.4643	0.3244
Combined systolic and diastolic heart failure, unspecified	0.8715	0.6269	0.8559	**0.8998**
Unspecified transient cerebral ischemia	0.6328	0.5479		**0.6945**
Atherosclerosis of native arteries of the extremities with rest pain	**0.9041**	0.7885	0.5801	0.5415
Abdominal aneurysm, ruptured	**0.9974**	0.9972		0.5881
Acute combined systolic and diastolic heart failure	0.9204	0.8526	**0.9426**	0.7184
Unspecified late effects of cerebrovascular disease	0.6250	**0.8097**		0.7283
Unspecified cerebrovascular disease	**0.8909**	0.5888		0.8324
Atherosclerosis of other specified arteries	0.9216	**0.9519**	0.7880	0.3573
Cerebral thrombosis with cerebral infarction	**0.9381**	0.9351		0.7420
Secondary cardiomyopathy, unspecified	**0.6790**	0.4789	0.6423	0.5308
Acute myocardial infarction of unspecified site, subsequent episode of care	**0.8799**	0.6348	0.8387	0.4825

Table 5.7: Predictive Performances of the Respiratory System

Disease name	**MTL**	**STL-highest**	**STL-related**	**No-Transfer**
Post-inflammatory pulmonary fibrosis	**0.9194**	0.8437	0.8694	0.8487
Pneumonia due to Pseudomonas	**0.9381**	0.9307	0.9267	0.9186
Chronic respiratory failure	**0.8879**	0.7741		0.7496
Acute edema of lung, unspecified	0.8358	0.8393	0.8332	**0.8662**
Chronic obstructive asthma with (acute) exacerbation	0.9866	**0.988**	0.9743	0.5019
Pneumonia due to other gram-negative bacteria	**0.9306**	0.9074	0.9271	0.9198
Bacterial pneumonia, unspecified	**0.8862**	0.8831	0.8773	0.8629
Pneumonia due to Klebsiella pneumoniae	**0.8858**	0.8685	0.8855	0.8179
Pneumococcal pneumonia [Streptococcus pneumoniae pneumonia]	0.8029	**0.8097**	0.7887	0.7833
Bronchiectasis without acute exacerbation	0.6873	**0.7980**	0.6988	0.7016
Empyema without mention of fistula	0.9858	**0.9959**	0.9888	0.9873
Methicillin resistant pneumonia due to Staphylococcus aureus	0.9291	**0.9600**		0.6498
Pulmonary congestion and hypostasis	0.6007	**0.7987**	0.5504	0.6108
Unspecified sinusitis (chronic)	0.586	**0.6584**	0.6340	0.6365
Edema of larynx	0.8221	**0.8302**		0.5535
Malignant pleural effusion	**0.9767**	0.9266	0.9842	0.8537
Acute bronchitis	0.7482	**0.8139**		0.4523
Asbestosis	**0.7506**	0.7375	0.4252	0.4262
Acute upper respiratory infections of unspecified site	0.9289	**0.9380**	0.8356	0.5727
Abscess of lung	0.8882	**0.9557**	0.9505	0.7781
Unilateral paralysis of vocal cords or larynx, partial	**0.809**	0.6864		0.4811
Empyema with fistula	0.9981	**0.9984**	0.9879	0.9875
Stenosis of larynx	**0.9979**	0.9976		0.9943

Table 5.8: Predictive Performances of the Genitourinary System

Disease name	**MTL**	**STL-highest**	**STL-related**	**No-Transfer**
Hematuria	**0.8609**	0.8168	0.7640	0.8124
Hydronephrosis	0.9513	**0.9690**		0.9690
Chronic kidney disease, Stage IV (severe)	0.8124	**0.8687**	0.5444	0.3528
Hypertrophy (benign) of prostate with urinary obstruction and other lower urinary tract symptoms (LUTS)	0.7134	0.7252	**0.7542**	0.6236
Neurogenic bladder NOS	**0.6845**	0.5690	0.6133	0.4658
Hematuria, unspecified	**0.7362**	0.6404	0.7012	0.6723
Calculus of kidney	**0.8299**	0.6291	0.6624	0.7257
Gross hematuria	0.859	0.8347	**0.8857**	0.8100
Secondary hyperparathyroidism (of renal origin)	0.7201	**0.7595**	0.5630	0.5462
Acute pyelonephritis without lesion of renal medullary necrosis	0.9534	**0.9710**	0.9528	0.9218
Calculus of ureter	0.9925	**0.9984**	0.5979	0.4971
Cyst of kidney, acquired	0.4814	**0.7061**	0.4498	0.5582
Hypertrophy (benign) of prostate	0.8882	0.4491	0.9069	**0.9320**
Pyelonephritis, unspecified	**0.9189**	0.8562	0.7573	0.5235
Nephritis and nephropathy, not specified as acute or chronic, with unspecified pathological lesion in kidney	0.7017	0.7093		**0.7848**
Vascular disorders of kidney	0.718	0.7303	0.7963	**0.9842**
Chronic glomerulonephritis in diseases classified elsewhere	0.2514	0.0676	0.1894	**0.5203**

When it comes to the comparison between either one of the three pre-trainings with *No-transfer*, we find that nearly 86% phenotypes perform better with the pre-training. The *MTL* outperforms *No-transfer* in 28 circulatory diseases, 18 respiratory diseases, and 11 genitourinary diseases. Among 78 diseases, the biggest AUC improvement from the pre-training to *No-transfer* is 0.5946 in the circulatory organ system with atherosclerosis of other specified arteries.

5.3.4.3 The Effectiveness of MTL

We show the phenotype counts where the *MTL* performs better than one of the baselines in Table 5.9.

Table 5.9: Comparisons of MTL with Baselines

# phenotypes	MTL & STL-highest		MTL & STL-related		MTL & Target	
	MTL > STL-highest	MTL < STL-highest	MTL > STL-related	MTL < STL-related	MTL > Target	MTL < Target
Circulatory	29	9	27	11	28	10
Respiratory	10	13	15	8	18	5
Genitourinary	8	9	11	6	11	6
Total	**47**	31	**53**	25	**57**	21

We find that the number of phenotypes for which *MTL* performs better than the other baseline is generally higher than the number of phenotypes for which *MTL* performs worse. Namely, among 78 phenotypes, 47 (60 %) have higher AUCs when using *MTL* over *STL-highest*, 53 (68 %) have higher AUCs when using *MTL* over *STL-related*, and 57 phenotypes (73 %) have higher AUCs when using *MTL* over *No-transfer*. The exact highest improvement with *MTL* over the best of the other three baselines is 0.2063 in the circulatory organ system, shown in Table 5.6.

Pre-training with *MTL* consistently and often enhanced performance to a large extent compared to pre-training with *STL*. Table 5.10 calculates the number of phenotypes where each of the methods achieves the best performance. We notice that *MTL* has the highest number of phenotypes that achieve the best. Specifically, 33 phenotypes (42 %) has the best AUCs with *MTL*.

Table 5.10: Number of Phenotypes for Best Performances and Tolerable Cases

Methods	**# Best** (%)	**# Within 90% of the best if not the best**	**# Total tolerable cases**. Including the best and within 90%. Sum of the two left columns.
MTL	**33** (**42**%)	**37**	**70** (**90**%)
STL-highest	25 (32%)	36	61 (78%)
STL-related	9 (11%)	44	53 (68%)
No-transfer	11 (14%)	27	38 (48%)

$$\text{Best performance } \% = \frac{\text{\# Best performance}}{\text{\# total phenotypes}} \tag{5.2}$$

$$\text{Tolerable performance } \% = \frac{\text{\# Within 90\% of the best}}{\text{\# total phenotypes}} \tag{5.3}$$

While the method may not perform optimally, we consider it acceptable if it performs within 90% of optimal level because performance gaps have no real impact on clinical practice, which we refer to as tolerable cases. We calculate such cases for four methods (in Table 5.10), meaning that when the method does not achieve the best result, the number of phenotypes is still within 90% of the best. The second column is the number of phenotypes that are still within 90% of the

best performance when the method does not perform the best. The last column is the total tolerable case, including the best performance and within 90% of the best performance. It is calculated using the total of the two left columns. We see that *MTL* achieves the highest majority percentage of tolerable cases with nearly 90%, which represents very few phenotypes (less than 10%) where *MTL* does not have a fair performance.

Finally, we calculate the average mean squared error (Avg-MSE) on the test set for each method across three organ systems, and the values are shown in the Table 5.11. The average mean squared error is defined as the difference between the estimates and the actual values, so the Avg-MSEs and predictive performances are inversely proportional. As a result, we see that across all systems, the Avg-MSE of *MTL* is the smallest of the other three baselines.

Table 5.11: Average Mean Squared Error across Organ Systems

	MTL	**STL-high**	**STL-related**	**No-transfer**
Circulatory	**0.0368**	0.0488	0.0477	0.0490
Respiratory	**0.0388**	0.0461	0.0397	0.0422
Genitourinary	**0.0384**	0.0431	0.0447	0.0500

In general, *MTL* has the highest proportion of the best performance, the highest proportion of tolerable cases, and also the smallest Avg-MSE. As expressly stated, when *MTL* performs worse than the baseline, it is still quite close, but when it outperforms the baseline, it is often substantially better. These experiment results support our assumption that *MTL* pre-training of high-prevalence phenotypes improves prediction performance on low-prevalence phenotypes in a robust and stable manner.

5.3.5 Discussion

In this section, with multi-task pre-training and fine-tuning, we show the feasibility of learning generic and portable patient language representations from medical notes. The pre-trained models learn the five most common phenotypes with multi-task learning for three organ systems. We further fine-tuned the organ-specific pre-trained models by applying them to a variety of low-prevalence phenotypes with sparse data. The findings are promising in that *MTL* pre-trained models consistently outperform baselines of both *STL* pre-trained models and *No-transfer* to predict low-prevalence phenotypes. Also, the *MTL* pre-training improves learning efficiency, because it only has to be pre-trained once. *MTL* methods are more efficient than either of the two *STL* methods, even though the *STL-related* phenotype is the most clinically relevant phenotype that is correlated to the target phenotype. This validates our prediction that the *MTL* model facilitates generalization by leveraging hidden information between tasks.

The predictive results of low-prevalence phenotypes have been consistently improved with MTL pre-training. We assume the reason is that source tasks in pre-training are based on the most common phenotypes in each organ system, enabling the model to learn jointly within the organ system. These pre-trained models for given organ systems are capable of learning intricate semantics through the integration of several phenotypes representing distinct causal factors and diverse surroundings. As a result, they are equipped to deal with complex and unexpected scenarios. For the sake of simplicity, five phenotypes per organ system are selected. In the future, we will increase the number of high-prevalence phenotypes (i.e., more than

five) in the source task to cover the majority of patients.

When considering the optimal way to adjust the pre-trained model for certain fine-tuning tasks, the feature extraction method is another alternative for this paradigm. However, instead of adapting the whole parameters like in pre-training and fine-tuning, the feature extraction method only extracts a static vector representation from intermediate layers of the pre-trained model. The vector can be extracted from any layer, such as the last multiple hidden layer or weighted combinations of layers. The vectors would then be fed as input layers to the downstream target task.Ever since the development of pre-trained language models, a few open-domain studies have evaluated these two approaches. Devlin et al. (2019) compared a few types of feature extraction and found that all of them performed worse than direct fine-tuning of BERT models. We agree that feature extraction offers a few advantages over the fine-tuning approach, most notably with regards to computing resources. With simple models, it takes far less time to extract a vector and use it in multiple downstream tasks than it does to rebuild a large model and fine-tune its parameters on another task.

Nevertheless, we believe that fine-tuning is required to enhance performance for our specialized downstream tasks. Because the vectors retrieved using low-dimensional (often hundreds or thousands) embeddings are not discriminative for the new task, which may be improved by fine-tuning with hyper-parameters that are specific to the new task. While choosing the feature extraction approach, if the model of the downstream task is still a large model (i.e., a deep neural network), the training may still be computationally intensive. The efficiency gains come from

only extracting the features once, rather than training the new task. Fine-tuning alters the parameters dynamically to make them more customized for the downstream task and also enables the model to target a general-purpose model for diverse tasks. Therefore, even considering the trade-off between time and performance, we still conclude that fine-tuning is the best way to build a representation for our situation.

CHAPTER 6

CONCLUSION

6.1 Thesis Overview

In this thesis, I discussed building machine learning models from words, documents, and finally patients, notably in terms of how to build representation learning to improve patient-level prediction. I believe that developing deep representation learning methods for distilling enormous amounts of heterogeneous data into patient-level representations is important and will in turn improve evidence-based clinical understanding. It is critical to consider various linguistic components in natural language. Such results have implications beyond the immediate context of predictions. I anticipate that this will be a starting point for future NLP-based phenotyping methods that develop neural network-enhanced patient-level representations to strengthen clinical predictions.

In CHAPTER 2, I reviewed and discussed the current state and challenges pertinent to representation learning in NLP and also patient representation learning. I concluded that deep representation learning has evolved into a great number of new approaches to modeling patient data. Deep patient representation learning is a feasible and promising route to develop effective, reliable, and specific representations. By incorporating cutting-edge learning methods into the neural network, patient representation learning aims to alleviate many of the challenges of EHR data and facilitate patient-oriented care. I expect that sophisticated techniques for learning

effective patient representations will continue to evolve, and these representations will potentially receive more attention in clinical predictions.

In CHAPTER 3, I developed a frame-based NLP system to extract cancer-relation terms with a Bi-LSTM-CRF model. The model was also initialized with different word embeddings along with the character embeddings as the input representations. The prediction obtains promising performance results, and the best F1-score of the lexical unit identification reaches 96.33% and that of the element classification gets 93.02%. Primarily, this proves the feasibility of developing a frame-based NLP system to extract cancer information from clinical notes. Ultimately, I hope to integrate all important cancer-related information and extract the information from one superior system. In the second section, I assessed the performance of different word embedding approaches on four clinical concept extraction shared-tasks. I compared conventional word embeddings with advanced language model-based embeddings. Additionally, I evaluated the results of pre-trained clinical domain embeddings against the off-the-shelf released embeddings. The effectiveness of contextual embeddings over conventional word representations is shown in the majority of tasks. Contextual embeddings also convey semantic meaning that conventional word representations fail to take into consideration. In the end, these findings further show the value of pre-training on clinical texts, which outperform the released models, and more importantly, I achieved the new state-of-the-art results across all tasks.

In CHAPTER 4, I presented a series of deep neural networks to encode different aspects of clinical notes, starting from hierarchical, contextual, and lastly, longitu-

dinal data. In the end, I combined these three characteristics together, and proposed the Hierarchical Transformer Network, which is developed to efficiently process large-scale clinical notes with sequential and hierarchical structures. The network considers the temporal relations between notes as well as the hierarchical structure. I evaluated the network to predict standard clinical outcomes, including in-hospital mortality and ten common phenotypes. My experimental results showed that the Hierarchical Transformer Network outperforms strong baselines in AUC, PRC, and F1-score for both predictions. I also experimented with extensive ablation studies on the proposed model to achieve robust and effective training with limited computer resources.

In CHAPTER 5, to obtain generic and transferable patient language representations from clinical notes, I developed a multi-task pre-training and fine-tuning (*MTL*) framework. This *MTL* framework provides a pipeline to train a model with multi-task learning of phenotypes and to continually fine-tune on low-prevalence phenotypes. I evaluated and fine-tuned MTL models for 38 circulatory phenotypes, 23 respiratory phenotypes, and 17 genitourinary phenotypes. As a result, *MTL* consistently improves the performance and learning efficiency compared to the other three baselines, including two single-task pre-trainings. All the experiment findings concluded that this *MTL* framework is a robust and efficient method for developing generalized and transferable patient language representations. Eventually, I hope that this pre-training and fine-tuning framework will be utilized to develop comprehensive medical language representations from diverse free-text sources.

6.2 Significance and Contribution

This dissertation provides significant and valuable contributions to the clinical NLP community. The main contributions of this thesis are as follows:

- Firstly, I addressed issues and complexities in NLP for medicine, including clinical text classification and modeling. I identified and categorized the core limitations of processing large-scale clinical notes into three types of representations: contextual information, hierarchical structure, and longitudinal sequence data.

- Following the first step, I developed state-of-the-art deep learning models to process a series of clinical notes on long-term dependency. I demonstrated the importance of neural networks developed for clinical notes on the predictions of standard clinical outcomes.

- I pioneered a new transfer learning framework with pre-training and fine-tuning to achieve supervised knowledge transfer in a model-wise manner, instead of simple feature extraction. These patient language representations can be applied to a wide variety of low-prevalence phenotypes that are typically challenging to achieve optimal results for.

- By building end-to-end projections between unstructured clinical notes and structured EHR data, I made extensive use of data and mitigated human annotation efforts. More importantly, this kind of mapping is capable of encoding medical knowledge into neural networks through neural representations.

- Last but not least, I initiated a new paradigm for NLP with a patient-level representation focus in order to meet the requirements of patient-level research problems, which compensates for task-specific or problem-specific representations. As significant volumes of knowledge regarding patients and research evidence are encapsulated in the form of free text, and with the constant development of deep language modeling, I believe clinical NLP holds particular promise for improving evidence-based and patient-oriented clinical research.

6.3 Limitations and Future Directions

This dissertation is subject to several limitations, and there is abundant room for further progress.

First, these findings are limited to phenotypes in the multi-task setting, and the findings may not generalize to all clinical outcomes. I hypothesize that incorporating different clinical outcomes appropriately into multi-task settings may improve the generalizability of transfer learning. In spite of my investigation into the advantages of transferable patient representations, I expect future study on the design of multiple tasks in pre-training. For example, it would be illuminating to establish the interaction between the number of tasks in a source task and the target task performance. I hypothesize that as the number of tasks grows, the pre-trained model becomes progressively enriched with comprehensive knowledge regarding the patient, which potentially enhances the prediction of target tasks. This knowledge transfer results in an effective model that not only is suitable to predicting across multiple tasks but also predicts with higher accuracy than training separate mod-

els for each of the individual tasks. However, on the other hand, training a model on too many tasks may downgrade model capacity because this might be the case when the tasks are not related. This would be an interesting trade-off between the complexity and efficacy of source tasks. I will extend the current experiments to construct more powerful source tasks with different numbers and diverse types of tasks. For example, I intend to implement a more recent and efficient method called Task Affinity Groupings (Fifty et al., 2021) to identify the types and numbers of tasks that should be jointly trained in multi-task scenarios.

We will also integrate the existing knowledge of rare diseases with this current experimental design to investigate actual rare disease phenotyping. When it comes to identifying uncommon diseases, prior studies have typically encountered difficulties due to inadequate diagnosis rates and a limited population sample. The advent of secondary EHR data has enlarged the possibilities for expanding understanding of these diseases. Many rare diseases may potentially benefit from advanced data-driven methods to exploit and synthesize multi-source data, so data-driven methods have been developed to overcome the challenges of classifying rare diseases. For instance, a few preliminary studies sought to leverage EHR data with data-driven methods to learn more about rare diseases (Jia et al., 2018; Schaefer et al., 2020). Garcelon et al. (2018) used the TF-IDF from NLP methods to extract clinical concepts from patient clinical notes for RETT syndrome to enrich the knowledge of the current phenotyping description. Shen et al. (2019) developed a series of data-driven methods based on graph convolutional networks (Shen et al., 2020a) and ontology embeddings (Shen et al., 2020b) to augment rare disease knowledge. In

this thesis, I concentrated on low-prevalence phenotypes (namely, the number of patients matching the phenotype is small) in a particular cohort of patients, instead of on rare diseases as defined by the NIH definition. These results though, I believe, demonstrated the tremendous potential for integrating disparate sources of information and for establishing meaningful conceptions of rare diseases. As a result, incorporating the knowledge base of rare diseases will be one of my upcoming focuses.

In the end, through my investigation, I realized that transfer learning in biomedical documents is still at its very early stage. Unlike in clinical image recognition (e.g., detection of diabetic retinopathy from fundus images) where transfer learning has already been widely and successfully applied, transfer learning for biomedical documents is currently at its starting point and many interesting open questions still exist. I assume the relatively slow adoption of transfer learning in clinical NLP is because clinical or biomedical documents are still under-utilized, which may be due to the lack of labeled data or the incapability of dealing with large amounts of texts.

Another reason would be the gap between the capabilities of neural networks to distill knowledge and how useful that knowledge would be, whether it actually meets the research needs. One may have very effective NLP models for one type of text, such as literature, but this type of model may not work well on another type of text, for example, clinical trials. Even if within the same type of context, how to identify an appropriate source task in the pre-training so that it would be beneficial to the target task, is still quite challenging. The optimal architecture should be able to capture a broad and comprehensive spectrum of information that is hidden in

unstructured resources.

Overall, the advancements made in this area over the past four years are very encouraging, and it has been fortunate for me to have the opportunity to make a contribution to this area. Meanwhile, I feel certain that there is still much to learn about clinical notes and that many open questions remain unanswered. We need to look at the science behind what has been addressed, rather than simply text modeling, to move forward to the next level of actual comprehension. Additionally, I have strove to inspire other professionals to continually adapt neural networks to clinical domains or tasks. I hope to further improve the model capability of understanding clinical notes and that the above directions will be investigated and expanded in potential approaches.

References

Adhikari, A., Ram, A., Tang, R., and Lin, J. (2019). DocBERT: BERT for document classification. *arXiv*, 1904.08398.

Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jindi, D., Naumann, T., and McDermott, M. (2019). Publicly Available Clinical BERT Embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78.

Amatriain, X. (2020). NLP & Healthcare: Understanding the language of medicine.

Ashish, N., Dahm, L., and Boicey, C. (2014). University of california, irvine–pathology extraction pipeline: The pathology extraction pipeline for information extraction from pathology reports. *Health informatics journal*, 20(4):288–305.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Banerjee, I., Ling, Y., Chen, M. C., Hasan, S. A., Langlotz, C. P., Moradzadeh, N., Chapman, B., Amrhein, T., Mong, D., Rubin, D. L., Farri, O., and Lungren, M. P. (2019). Comparative effectiveness of convolutional neural network (cnn) and recurrent neural network (rnn) architectures for radiology text report classification. *Artificial Intelligence in Medicine*, 97:79 – 88.

Baumel, T., Nassour-Kassis, J., Cohen, R., Elhadad, M., and Elhadad, N. (2018). Multi-label classification of patient notes: case study on icd code assignment. In *Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 409–416.

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A re-

view and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Browne, A. C., McCray, A. T., and Srinivasan, S. (2000). The specialist lexicon. *National Library of Medicine Technical Reports*, pages 18–21.

Cao, L., Li, L., Zheng, J., Fan, X., Yin, F., Shen, H., and Zhang, J. (2018). Multi-task neural networks for joint hippocampus segmentation and clinical score regression. *Multimedia Tools and Applications*, 77(22):29669–29686.

Caruana, R., Baluja, S., and Mitchell, T. (1995). Using the future to" sort out" the present: Rankprop and multitask learning for medical risk evaluation. *Advances in Neural Information Processing Systems*, 8:959–965.

Cavnar, W. B., Trenkle, J. M., et al. (1994). N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*, volume 161175. Citeseer.

Chalapathy, R., Zare Borzeshi, E., and Piccardi, M. (2016). Bidirectional lstm-crf for clinical concept extraction. In *Clinical Natural Language Programming Workshop*. ClinicalNLP.

Che, Z., Kale, D., Li, W., Bahadori, M. T., and Liu, Y. (2015). Deep computational phenotyping. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 507–516.

Cheng, L. T., Zheng, J., Savova, G. K., and Erickson, B. J. (2010). Discerning tumor status from unstructured mri reports—completeness of information in existing

reports and utility of automated natural language processing. *Journal of digital imaging*, 23(2):119–132.

Cheng, Y., Wang, F., Zhang, P., and Hu, J. (2016). Risk prediction with electronic health records: A deep learning approach. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 432–440. SIAM.

Chi, E. C. and Kolda, T. G. (2012). On tensors, sparsity, and nonnegative factorizations. *SIAM Journal on Matrix Analysis and Applications*, 33(4):1272–1299.

Cho, K., Van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Choi, E., Bahadori, M. T., Schuetz, A., Stewart, W. F., and Sun, J. (2016a). Doctor ai: Predicting clinical events via recurrent neural networks. In *Machine learning for healthcare conference*, pages 301–318. PMLR.

Choi, E., Bahadori, M. T., Song, L., Stewart, W. F., and Sun, J. (2017). Gram: graph-based attention model for healthcare representation learning. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 787–795.

Choi, E., Xu, Z., Li, Y., Dusenberry, M., Flores, G., Xue, E., and Dai, A. (2020). Learning the graphical structure of electronic health records with graph convolutional transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 606–613.

Choi, Y., Chiu, C. Y.-I., and Sontag, D. (2016b). Learning low-dimensional representations of medical concepts. *AMIA Summits on Translational Science Proceedings*, 2016:41.

Chowdhuri, S., Pankaj, T., and Zipser, K. (2019). Multinet: Multi-modal multi-task learning for autonomous driving. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1496–1504. IEEE.

Clark, K., Luong, M.-T., Le, Q. V., and Manning, C. D. (2020). ELECTRA: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Coden, A., Savova, G., Sominsky, I., Tanenblatt, M., Masanz, J., Schuler, K., Cooper, J., Guan, W., and De Groen, P. C. (2009). Automatically extracting cancer disease characteristics from pathology reports into a disease knowledge representation model. *Journal of biomedical informatics*, 42(5):937–949.

Cohan, A., Fong, A., Goharian, N., and Ratwani, R. M. (2017a). A neural attention model for categorizing patient safety events. *ArXiv*, abs/1702.07092.

Cohan, A., Fong, A., Ratwani, R. M., and Goharian, N. (2017b). Identifying harm events in clinical care through medical narratives. In *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, ACM-BCB, pages 52–59.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

Crichton, G., Pyysalo, S., Chiu, B., and Korhonen, A. (2017). A neural network multi-task learning approach to biomedical named entity recognition. *BMC bioinformatics*, 18(1):1–14.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., and Salakhutdinov, R. (2019). Transformer-XL: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

D'Avolio, L. W., Litwin, M. S., Rogers Jr, S. O., and Bui, A. A. (2008). Facilitating clinical outcomes assessment through the automated identification of quality measures for prostate cancer surgery. *Journal of the American Medical Informatics Association*, 15(3):341–348.

Demner-Fushman, D., Shooshan, S. E., Rodriguez, L., Aronson, A. R., Lang, F., Rogers, W., Roberts, K., and Tonning, J. (2018). A dataset of 200 structured product labels annotated for adverse drug reactions. *Scientific data*, 5(1):1–8.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Denny, J. C., Choma, N. N., Peterson, J. F., Miller, R. A., Bastarache, L., Li, M., and Peterson, N. B. (2012). Natural language processing improves identification of colorectal cancer testing in the electronic medical record. *Medical Decision Making*, 32(1):188–197.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Ding, D. Y., Simpson, C., Pfohl, S., Kale, D. C., Jung, K., and Shah, N. H. (2018). The effectiveness of multitask learning for phenotyping with electronic health records data. In *BIOCOMPUTING 2019: Proceedings of the Pacific Symposium*, pages 18–29. World Scientific.

Dligach, D., Afshar, M., and Miller, T. (2019). Toward a clinical text encoder: pretraining for clinical natural language processing with applications to substance

misuse. *Journal of the American Medical Informatics Association*, 26(11):1272–1278.

Dong, D., Wu, H., He, W., Yu, D., and Wang, H. (2015). Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1723–1732.

Du, J., Chen, Q., Peng, Y., Xiang, Y., Tao, C., and Lu, Z. (2019). ML-Net: multi-label classification of biomedical texts with deep neural networks. *Journal of the American Medical Informatics Association*, 26(11):1279–1285.

Elhadad, N., Pradhan, S., Gorman, S., Manandhar, S., Chapman, W., and Savova, G. (2015). Semeval-2015 task 14: Analysis of clinical text. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 303–310.

Elhoseiny, M., El-Gaaly, T., Bakry, A., and Elgammal, A. (2016). A comparative analysis and study of multiview cnn models for joint object categorization and pose estimation. In *International Conference on Machine learning*, pages 888–897. PMLR.

Fa, R., Cozzetto, D., Wan, C., and Jones, D. T. (2018). Predicting human protein function with multi-task deep neural networks. *PloS one*, 13(6):e0198216.

Fifty, C., Amid, E., Zhao, Z., Yu, T., Anil, R., and Finn, C. (2021). Efficiently identifying task groupings for multi-task learning. In *Thirty-Fifth Conference on Neural Information Processing Systems*.

Futoma, J., Hariharan, S., and Heller, K. (2017). Learning to detect sepsis with

a multitask gaussian process rnn classifier. In *International Conference on Machine Learning*, pages 1174–1182. PMLR.

Gao, S., Young, M. T., Qiu, J. X., Yoon, H.-J., Christian, J. B., Fearn, P. A., Tourassi, G. D., and Ramanthan, A. (2017). Hierarchical attention networks for information extraction from cancer pathology reports. *Journal of the American Medical Informatics Association*, 25(3):321–330.

Garcelon, N., Neuraz, A., Salomon, R., Bahi-Buisson, N., Amiel, J., Picard, C., Mahlaoui, N., Benoit, V., Burgun, A., and Rance, B. (2018). Next generation phenotyping using narrative reports in a rare disease clinical data warehouse. *Orphanet journal of rare diseases*, 13(1):1–11.

Gehrmann, S., Dernoncourt, F., Li, Y., Carlson, E. T., Wu, J. T., Welt, J., Foote Jr, J., Moseley, E. T., Grant, D. W., Tyler, P. D., et al. (2018). Comparing deep learning and concept extraction based methods for patient phenotyping from clinical narratives. *PloS one*, 13(2):e0192360.

Gridach, M. (2017). Character-level neural network for biomedical named entity recognition. *Journal of biomedical informatics*, 70:85–91.

Grover, A. and Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864.

Habibi, M., Weber, L., Neves, M., Wiegandt, D. L., and Leser, U. (2017). Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48.

Harkema, H., Chapman, W. W., Saul, M., Dellon, E. S., Schoen, R. E., and Mehrotra, A. (2011). Developing a natural language processing application for mea-

suring the quality of colonoscopy procedures. *Journal of the American Medical Informatics Association*, 18(Supplement_1):i150–i156.

Harutyunyan, H., Khachatrian, H., Kale, D. C., Ver Steeg, G., and Galstyan, A. (2019). Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):1–18.

He, H., Henderson, J., and Ho, J. C. (2019). Distributed tensor decomposition for large scale health analytics. In *The World Wide Web Conference*, pages 659–669.

Hinton, G. E. et al. (1986). Learning distributed representations of concepts. In *Proceedings of the eighth annual conference of the cognitive science society*, volume 1, page 12. Amherst, MA.

Ho, J. C., Ghosh, J., and Sun, J. (2014). Marble: high-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 115–124.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Huang, K., Altosaar, J., and Ranganath, R. (2019). Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*.

Imler, T. D., Morea, J., Kahi, C., and Imperiale, T. F. (2013). Natural language processing accurately categorizes findings from colonoscopy and pathology reports. *Clinical Gastroenterology and Hepatology*, 11(6):689–694.

Ive, J., Gkotsis, G., Dutta, R., Stewart, R., and Velupillai, S. (2018). Hierarchical neural model with attention mechanisms for the classification of social media text

related to mental health. In *Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*, pages 69–77.

Jagannatha, A. N. and Yu, H. (2016a). Bidirectional rnn for medical event detection in electronic health records. In *Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting*, volume 2016, page 473. NIH Public Access.

Jagannatha, A. N. and Yu, H. (2016b). Structured prediction models for rnn based sequence labeling in clinical text. In *Proceedings of the conference on empirical methods in natural language processing. conference on empirical methods in natural language processing*, volume 2016, page 856. NIH Public Access.

Jia, J., An, Z., Ming, Y., Guo, Y., Li, W., Liang, Y., Guo, D., Li, X., Tai, J., Chen, G., et al. (2018). eram: encyclopedia of rare disease annotations for precision medicine. *Nucleic acids research*, 46(D1):D937–D943.

Johnson, A. E., Pollard, T. J., Shen, L., Li-Wei, H. L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. (2016). Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., and Levy, O. (2020). SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Kemp, J., Rajkomar, A., and Dai, A. M. (2019). Improved hierarchical patient classification with language model pretraining over clinical notes. *arXiv preprint arXiv:1909.03039*.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In Moschitti, A., Pang, B., and Daelemans, W., editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014,*

October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1746–1751. ACL.

Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM review*, 51(3):455–500.

Kumar, A. and Daumé III, H. (2012). Learning task grouping and overlap in multi-task learning. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1723–1730.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of NAACL-HLT*, pages 260–270.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). AlBERT: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L. (1989). Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2020). Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Li, F., Zhang, M., Fu, G., and Ji, D. (2017). A neural joint model for entity and relation extraction from biomedical text. *BMC bioinformatics*, 18(1):1–11.

Li, S., Liu, Z.-Q., and Chan, A. B. (2014). Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 482–489.

Li, Y., Rao, S., Solares, J. R. A., Hassaine, A., Ramakrishnan, R., Canoy, D., Zhu, Y., Rahimi, K., and Salimi-Khorshidi, G. (2020). BEHRT: transformer for electronic health records. *Scientific reports*, 10(1):1–12.

Liu, C., Wang, F., Hu, J., and Xiong, H. (2015). Temporal phenotyping from longitudinal electronic health records: A graph based framework. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 705–714.

Liu, P., Qiu, X., and Huang, X.-J. (2017a). Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–10.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Liu, Z., Lin, Y., and Sun, M. (2020). *Representation learning for natural language processing*. Springer Nature.

Liu, Z., Yang, M., Wang, X., Chen, Q., Tang, B., Wang, Z., and Xu, H. (2017b). Entity recognition from clinical texts via recurrent neural network. *BMC medical informatics and decision making*, 17(2):53–61.

Lopez-Martinez, D. and Picard, R. (2017). Multi-task neural networks for personalized pain recognition from physiological signals. In *2017 Seventh International*

Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW), pages 181–184. IEEE.

Luo, Y. (2017). Recurrent neural networks for classifying relations in clinical notes. *Journal of biomedical informatics*, 72:85–95.

Luo, Y., Cheng, Y., Uzuner, Ö., Szolovits, P., and Starren, J. (2018). Segment convolutional neural networks (seg-cnns) for classifying relations in clinical notes. *Journal of the American Medical Informatics Association*, 25(1):93–98.

Ma, F., You, Q., Xiao, H., Chitta, R., Zhou, J., and Gao, J. (2018). Kame: Knowledge-based attention model for diagnosis prediction in healthcare. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 743–752.

Martinez, D., Cavedon, L., and Pitson, G. (2013). Stability of text mining techniques for identifying cancer staging. In *Louhi, The 4th International Workshop on Health Document Text Mining and Information Analysis, NICTA, Canberra, Australia*.

Martinez, D. and Li, Y. (2011). Information extraction from pathology reports in a hospital setting. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1877–1882.

McCann, B., Keskar, N. S., Xiong, C., and Socher, R. (2018). The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

McCowan, I. A., Moore, D. C., Nguyen, A. N., Bowman, R. V., Clarke, B. E., Duhig, E. E., and Fry, M.-J. (2007). Collection of cancer stage data by classifying free-text medical reports. *Journal of the American Medical Informatics Association*, 14(6):736–745.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Miotto, R., Li, L., Kidd, B. A., and Dudley, J. T. (2016). Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. *Scientific reports*, 6(1):1–10.

Miwa, M. and Bansal, M. (2016). End-to-end relation extraction using lstms on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1105–1116.

Mullenbach, J., Wiegreffe, S., Duke, J., Sun, J., and Eisenstein, J. (2018). Explainable prediction of medical codes from clinical text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1101–1111.

Napolitano, G., Fox, C., Middleton, R., and Connolly, D. (2010). Pattern-based information extraction from pathology reports for cancer registration. *Cancer Causes & Control*, 21(11):1887–1894.

Ngufor, C., Upadhyaya, S., Murphree, D., Kor, D., and Pathak, J. (2015). Multitask learning with selective cross-task transfer for predicting bleeding and other important patient outcomes. In *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–8. IEEE.

Niepert, M., Ahmed, M., and Kutzkov, K. (2016). Learning convolutional neural

networks for graphs. In *International conference on machine learning*, pages 2014–2023. PMLR.

Nori, N., Kashima, H., Yamashita, K., Kunisawa, S., and Imanaka, Y. (2017). Learning implicit tasks for patient-specific risk modeling in icu. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Ou, Y. and Patrick, J. (2014). Automatic population of structured reports from narrative pathology reports. In *Proceedings of the Seventh Australasian Workshop on Health Informatics and Knowledge Management - Volume 153*, HIKM '14, page 41–50.

Pakhomov, S. V., Finley, G., McEwan, R., Wang, Y., and Melton, G. B. (2016). Corpus domain effects on distributional semantic modeling of medical terms. *Bioinformatics*, 32(23):3635–3644.

Pappagari, R., Zelasko, P., Villalba, J., Carmiel, Y., and Dehak, N. (2019). Hierarchical transformers for long document classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 838–844. IEEE.

Pasunuru, R. and Bansal, M. (2017). Multi-task video captioning with video and entailment generation. *arXiv preprint arXiv:1704.07489*.

Peng, Y., Yan, S., and Lu, Z. (2019). Transfer Learning in Biomedical Natural Language Processing: An Evaluation of BERT and ELMo on Ten Benchmarking Datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.

Ping, X.-O., Tseng, Y.-J., Chung, Y., Wu, Y.-L., Hsu, C.-W., Yang, P.-M., Huang, G.-T., Lai, F., and Liang, J.-D. (2013). Information extraction for tracking liver cancer patients' statuses: from mixture of clinical narrative report types. *TELEMEDICINE and e-HEALTH*, 19(9):704–710.

Popel, M. and Bojar, O. (2018). Training tips for the transformer model. 110(1):43–70. Publisher: Sciendo.

Pradhan, S., Elhadad, N., Chapman, W., Manandhar, S., and Savova, G. (2014). Semeval-2014 task 7: Analysis of clinical text. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 54–62.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018). Improving language understanding by generative pre-training.

Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Liu, P. J., Liu, X., Marcus, J., Sun, M., et al. (2018). Scalable and accurate deep learning with electronic health records. *NPJ Digital Medicine*, 1(1):1–10.

Ramos, J. et al. (2003). Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer.

Ramsundar, B., Kearnes, S., Riley, P., Webster, D., Konerding, D., and Pande, V. (2015). Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072*.

Rasmy, L., Xiang, Y., Xie, Z., Tao, C., and Zhi, D. (2021). Med-bert: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *NPJ digital medicine*, 4(1):1–13.

Razavian, N., Marcus, J., and Sontag, D. (2016). Multi-task prediction of disease onsets from longitudinal laboratory tests. In *Machine learning for healthcare conference*, pages 73–100. PMLR.

Rei, M. (2017). Semi-supervised multitask learning for sequence labeling. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2121–2130.

Roberts, K. (2016). Assessing the corpus size vs. similarity trade-off for word embeddings in clinical nlp. In *Proceedings of the Clinical Natural Language Processing Workshop (ClinicalNLP)*, pages 54–63.

Roberts, K., Si, Y., Gandhi, A., and Bernstam, E. (2018). A framenet for cancer information in clinical narratives: schema and annotation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Ruder, S. (2017). An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Samonte, M. J. C., Gerardo, B. D., Fajardo, A. C., and Medina, R. P. (2018). Icd-9 tagging of clinical notes using topical word embedding. In *Proceedings of the 2018 International Conference on Internet and e-Business*, pages 118–123.

Samonte, M. J. C., Gerardo, B. D., and Medina, R. P. (2017). Towards enhanced hierarchical attention networks in icd-9 tagging of clinical notes. In *Proceedings of the 3rd International Conference on Communication and Information Processing*, pages 146–150.

Savova, G. K., Tseytlin, E., Finan, S., Castine, M., Miller, T., Medvedeva, O., Harris, D., Hochheiser, H., Lin, C., Chavan, G., et al. (2017). Deepphe: a natural language processing system for extracting cancer phenotypes from clinical records. *Cancer research*, 77(21):e115–e118.

Schaefer, J., Lehne, M., Schepers, J., Prasser, F., and Thun, S. (2020). The use of machine learning in rare diseases: a scoping review. *Orphanet Journal of Rare Diseases*, 15(1):1–10.

Shen, F., Wen, A., and Liu, H. (2020a). Enrich rare disease phenotypic characterizations via a graph convolutional network based recommendation system. In *2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, pages 37–40. IEEE.

Shen, F., Wen, A., and Liu, H. (2020b). Subgrouping rare disease patients leveraging the human phenotype ontology embeddings. In *2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, pages 169–172. IEEE.

Shen, F., Zhao, Y., Wang, L., Mojarad, M. R., Wang, Y., Liu, S., and Liu, H. (2019). Rare disease knowledge enrichment through a data-driven approach. *BMC medical informatics and decision making*, 19(1):1–11.

Si, Y., Bernstam, E. V., and Roberts, K. (2021a). Generalized and transferable patient language representation for phenotyping with limited data. *Journal of Biomedical Informatics*, 116:103726.

Si, Y., Du, J., Li, Z., Jiang, X., Miller, T., Wang, F., Zheng, W. J., and Roberts, K. (2021b). Deep representation learning of patient data from electronic health records (ehr): A systematic review. *Journal of Biomedical Informatics*, 115:103671.

Si, Y. and Roberts, K. (2018). A frame-based nlp system for cancer-related information extraction. In *AMIA Annual Symposium Proceedings*, volume 2018, page 1524. American Medical Informatics Association.

Si, Y. and Roberts, K. (2019). Deep patient representation of clinical notes via multi-task learning for mortality prediction. *AMIA Summits on Translational Science Proceedings*, 2019:779.

Si, Y. and Roberts, K. (2020). Patient Representation Transfer Learning from Clinical Notes based on Hierarchical Attention Network. *AMIA Summits on Translational Science Proceedings*, 2020:597.

Si, Y. and Roberts, K. (2021). Hierarchical transformer networks for longitudinal clinical document classification. *arXiv preprint arXiv:2104.08444*.

Si, Y., Wang, J., Xu, H., and Roberts, K. (2019). Enhancing clinical concept extraction with contextual embeddings. *Journal of the American Medical Informatics Association*, 26(11):1297–1304.

Song, H., Rajan, D., Thiagarajan, J. J., and Spanias, A. (2018). Attend and diagnose: Clinical time series analysis using attention models. In *Thirty-second AAAI conference on artificial intelligence*.

Stein-O'Brien, G. L., Arora, R., Culhane, A. C., Favorov, A. V., Garmire, L. X., Greene, C. S., Goff, L. A., Li, Y., Ngom, A., Ochs, M. F., et al. (2018). Enter the matrix: factorization uncovers knowledge from omics. *Trends in Genetics*, 34(10):790–805.

Steinberg, E., Jung, K., Fries, J. A., Corbin, C. K., Pfohl, S. R., and Shah, N. H. (2021). Language models are an effective representation learning technique for electronic health record data. *Journal of Biomedical Informatics*, 113:103637.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). 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.

Stubbs, A., Kotfila, C., and Uzuner, Ö. (2015). Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/uthealth shared task track 1. *Journal of biomedical informatics*, 58:S11–S19.

Sun, W., Rumshisky, A., and Uzuner, O. (2013). Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5):806–813.

Sushil, M., Šuster, S., Luyckx, K., and Daelemans, W. (2018). Patient representation learning and interpretable evaluation using clinical notes. *Journal of biomedical informatics*, 84:103–113.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Svozil, D., Kvasnicka, V., and Pospichal, J. (1997). Introduction to multi-layer feed-forward neural networks. *Chemometrics and intelligent laboratory systems*, 39(1):43–62.

Taira, R. K., Soderland, S. G., and Jakobovits, R. M. (2001). Automatic structuring of radiology free-text reports. *Radiographics*, 21(1):237–245.

Tang, B., Chen, Q., Wang, X., Wu, Y., Zhang, Y., Jiang, M., Wang, J., and Xu, H. (2015). Recognizing disjoint clinical concepts in clinical text using machine learning-based methods. In *AMIA Annual Symposium Proceedings*, volume 2015, page 1184.

Tao, C., Filannino, M., and Uzuner, Ö. (2017). Prescription extraction using crfs and word embeddings. *Journal of biomedical informatics*, 72:60–66.

Taslaman, L. and Nilsson, B. (2012). A framework for regularized non-negative matrix factorization, with application to the analysis of gene expression data. *PloS one*, 7(11):e46331.

Toshniwal, S., Tang, H., Lu, L., and Livescu, K. (2017). Multitask learning with low-level auxiliary tasks for encoder-decoder based speech recognition. *arXiv preprint arXiv:1704.01631*.

Tran, T. and Kavuluru, R. (2017). Predicting mental conditions based on "history of present illness" in psychiatric notes with deep neural networks. *Journal of biomedical informatics*, 75:138–148.

Turc, I., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*.

Uzuner, Ö. (2009). Recognizing obesity and comorbidities in sparse data. *Journal of the American Medical Informatics Association*, 16(4):561–570.

Uzuner, Ö., South, B. R., Shen, S., and DuVall, S. L. (2011). 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Vanderwende, L., Xia, F., and Yetisgen-Yildiz, M. (2013). Annotating change of state for clinical events. In *Workshop on Events: Definition, Detection, Coreference, and Representation*, pages 47–51.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser,

Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y. (2017). Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103.

Wallach, H. M. (2006). Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pages 977–984.

Wang, F., Lee, N., Hu, J., Sun, J., and Ebadollahi, S. (2012a). Towards heterogeneous temporal clinical event pattern discovery: a convolutional approach. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 453–461.

Wang, F., Lee, N., Hu, J., Sun, J., Ebadollahi, S., and Laine, A. F. (2012b). A framework for mining signatures from event sequences and its applications in healthcare data. *IEEE transactions on pattern analysis and machine intelligence*, 35(2):272–285.

Wang, H., Zhang, W., Zeng, Q., Li, Z., Feng, K., and Liu, L. (2014a). Extracting important information from chinese operation notes with natural language processing methods. *Journal of biomedical informatics*, 48:130–136.

Wang, S., Ren, P., Chen, Z., Ren, Z., Ma, J., and de Rijke, M. (2019). Order-free medicine combination prediction with graph convolutional reinforcement learning. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1623–1632.

Wang, X., Wang, F., Hu, J., and Sorrentino, R. (2014b). Exploring joint disease risk prediction. In *AMIA annual symposium proceedings*, volume 2014, page 1180. American Medical Informatics Association.

Weegar, R. and Dalianis, H. (2015). Creating a rule based system for text mining of norwegian breast cancer pathology reports. In *Proceedings of the Sixth International Workshop on Health Text Mining and Information Analysis*, pages 73–78.

Wiens, J., Guttag, J., and Horvitz, E. (2016). Patient risk stratification with time-varying parameters: a multitask learning approach. *The Journal of Machine Learning Research*, 17(1):2797–2819.

Wilson, R. A., Chapman, W. W., DeFries, S. J., Becich, M. J., and Chapman, B. E. (2010). Automated ancillary cancer history classification for mesothelioma patients from free-text clinical reports. *Journal of pathology informatics*, 1.

Wu, J., Roy, J., and Stewart, W. F. (2010). Prediction modeling using ehr data: challenges, strategies, and a comparison of machine learning approaches. *Medical care*, pages S106–S113.

Wu, Y., Jiang, M., Xu, J., Zhi, D., and Xu, H. (2017). Clinical named entity recognition using deep learning models. In *AMIA Annual Symposium Proceedings*, volume 2017, page 1812. American Medical Informatics Association.

Xiang, Y., Xu, J., Si, Y., Li, Z., Rasmy, L., Zhou, Y., Tiryaki, F., Li, F., Zhang, Y., Wu, Y., et al. (2019). Time-sensitive clinical concept embeddings learned from large electronic health records. *BMC medical informatics and decision making*, 19(2):139–148.

Xu, H., Anderson, K., Grann, V. R., and Friedman, C. (2004). Facilitating cancer

research using natural language processing of pathology reports. In *MEDINFO 2004*, pages 565–569. IOS Press.

Xu, H., Zhengyan, Z., Ning, D., Yuxian, G., Xiao, L., Yuqi, H., Jiezhong, Q., Liang, Z., Wentao, H., Minlie, H., et al. (2021). Pre-trained models: Past, present and future. *arXiv preprint arXiv:2106.07139*.

Xu, J., Lee, H.-J., Ji, Z., Wang, J., Wei, Q., and Xu, H. (2017). Uth_ccb system for adverse drug reaction extraction from drug labels at tac-adr 2017. In *TAC*.

Xu, Y., Biswal, S., Deshpande, S. R., Maher, K. O., and Sun, J. (2018). Raim: Recurrent attentive and intensive model of multimodal patient monitoring data. In *Proceedings of the 24th ACM SIGKDD international conference on Knowledge Discovery & Data Mining*, pages 2565–2573.

Yala, A., Barzilay, R., Salama, L., Griffin, M., Sollender, G., Bardia, A., Lehman, C., Buckley, J. M., Coopey, S. B., Polubriaginof, F., et al. (2017). Using machine learning to parse breast pathology reports. *Breast cancer research and treatment*, 161(2):203–211.

Yang, K., Li, X., Liu, H., Mei, J., Xie, G., Zhao, J., Xie, B., and Wang, F. (2017). Tagited: Predictive task guided tensor decomposition for representation learning from electronic health records. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Yang, L., Zhang, M., Li, C., Bendersky, M., and Najork, M. (2020). Beyond 512 Tokens: Siamese Multi-depth Transformer-based Hierarchical Encoder for Long-Form Document Matching. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1725–1734.

Yang, X., Kim, S., and Xing, E. (2009). Heterogeneous multitask learning with

joint sparsity constraints. *Advances in neural information processing systems*, 22:2151–2159.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. (2016). Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. (2020). Big bird: Transformers for longer sequences. In *NeurIPS*.

Zhang, D., Thadajarassiri, J., Sen, C., and Rundensteiner, E. (2020). Time-Aware Transformer-based Network for Clinical Notes Series Prediction. In *Machine Learning for Healthcare Conference*, pages 566–588. PMLR.

Zhang, J., Gong, J., and Barnes, L. (2017). Hcnn: Heterogeneous convolutional neural networks for comorbid risk prediction with electronic health records. In *2017 IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*, pages 214–221. IEEE.

Zhang, X.-M., Liang, L., Liu, L., and Tang, M.-J. (2021). Graph neural networks and their current applications in bioinformatics. *Frontiers in Genetics*, 12.

Zhang, Y., Jin, R., and Zhou, Z.-H. (2010). Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1-4):43–52.

Zhang, Y., Wang, J., Tang, B., Wu, Y., Jiang, M., Chen, Y., and Xu, H. (2014). UTH_CCB: a report for semeval 2014–task 7 analysis of clinical text. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 802–806.

Zhang, Y. and Yang, Q. (2021). A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*.

Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., and Liu, Q. (2019). ERNIE: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129*.

Zhou, J., Wang, F., Hu, J., and Ye, J. (2014). From micro to macro: data driven phenotyping by densification of longitudinal electronic medical records. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 135–144.

Zhu, H., Paschalidis, I. C., and Tahmasebi, A. M. (2018). Clinical concept extraction with contextual word embedding. In *NIPS Machine Learning for Health Workshop*.