Extraction of Health Outcomes from Clinical Trial Abstracts

Micheal Abaho

A thesis submitted in accordance with the requirements of the University of Liverpool for the degree of Doctor in Philosophy.

November 2022

ACKNOWLEDGMENTS

A quiet review and reflection of all that has transpired in course of my PhD, humbles me and inevitably causes me to recognize a group of people without whom, completion of the PhD's deliverables including this thesis would have been quite harder than it was.

With great pleasure, I first and foremost thank the Almighty God. Without him, this would only be an imagination that would have never come to pass, his mercies, favor and love keep me going every single day of my life.

I would like to unequivocally appreciate the support, advice and remarkable insight of my primary supervisor, Prof. Danushka Bollegala. His profound knowledge and interest in the technical aspects pertinent to the research made it such an incredible privilege to work with him throughout the PhD.

Additionally, I also appreciate secondary supervisers, Prof. Paula R Williamson and Dr. Susanna Dodd for their involvement and contributions made in the build up to the completion of work in this thesis. Their expertise in the domain the thesis investigates was immense, particularly because they identified some of the gaps in the domain which present the problem the thesis confronts.

I would also like to acknowledge the advisory team of Prof Xiaowei Huang and Dr. Shan Luo for their constructive comments on the various propositions made during the course of the study.

Huge credit goes to the departments of Computer Science and Health Data Science in the university for providing the resources that enabled the PhD's activities and also making the whole experience worthwhile. Having been based in Computer Science, I was privileged to work with different lecturers in a Teaching Assistant role. The responsibility that comes along with the role was value addition to my research, and it has equipped me for future challenges at any academic or industrial level. Members of all research groups that I was apart of, particularly the NLP@Liv members, were outstanding in terms of sharing knowledge through weekly presentations on some of the latest trending research topics of interest to me.

Many many thanks to my family and friends for their availability, support and encouragement. My wife, Liz M, my mother Sarah N, my uncle Robert B, my family at large and all members of Hope Church Kensington, Liverpool. As I persevered, they advised, they encouraged, they motivated, they stood with me every step of the way. I sincerely express my gratitude towards all of them for all of this.

EXECUTIVE SUMMARY

How do clinicians determine whether a particular medical intervention actually works? Ideally, they would design and perform research studies in human beings (also called clinical trials) in order to compare the effects of the intervention against alternatives, after which they can generally establish its effectiveness and safety. To discover this effectiveness (evidence), Evidence-based Medicine (EBM) clinical practice recommends a framework for constructing clinical questions by carefully aligning them to four different components, i.e. Participants/Problem (P), Intervention (I), Comparison (C) and Outcome (O), all together abbreviated as PICO. Sample questions they may ask to obtain evidence include, *What is the primary problem and what are the patients' characteristics?* (e.g. *young children with acute febrile illness*), *What is the main intervention?* (e.g. *acetaminophen*)? *What is the main intervention compared to?* (e.g. *no intervention, ibuprofen*) and *What is the effect of the intervention?* (e.g. *fever*).

Unfortunately, the literature that summarises clinical trials is primarily disseminated in natural language articles which imposes a significant burden on clinicians to patiently read through them to pinpoint relevant evidence. The downside of this exercise is, it is time-consuming, costly and prone to human error especially because there is an astronomical volume of articles that are potential sources of evidence. To ensure effective and efficient healthcare decision making, clinicians would benefit from decision support or question answering systems that can search through abundant literature for evidence in a precise and timely manner. Driven by this requirement, this thesis addresses the problem clinicians face as articulated above.

The thesis pays maximum attention to the outcome (O) element introduced in the first paragraph. As applied to EBM, an outcome is a measurement or an observation used to capture and assess the effectiveness of treatment such as assessment of side effects (risk) or benefits. Automatic detection of outcomes from unstructured text would undoubtedly speed up access to evidence necessary in healthcare decision making, thereby improving public healthcare delivery. This thesis adopts and applies Natural Language Processing (NLP) techniques to automate the detection of outcomes from unstructured text. Despite outcome detection being an under researched subject, the thesis builds on prior efforts in detecting PICO elements to propose cost effective and optimised methods for outcome detection. The thesis also builds on the recent advancements in NLP such as transfer learning that enable the re-use of pre-trained language models

(PLMs) in downstream tasks different from the ones they were originally trained for.

The thesis designs and implements a framework that automatically corrects incorrectly captured annotations of outcomes, thereby improving the quality of the crowdsourced annotations. It further presents a flexible and unsupervised label denoiser which relies on a semantic similarity based approach to align weak (noisy) labels to standardised labels. An evaluation of these methods lead to substantial gains in performance of the outcome detection tasks. The thesis introduces "EBM-COMET", a dataset in which 300 Randomised Clinical Trial (RCT) PubMed abstracts are expertly annotated for clinical outcomes. Unlike prior related datasets that use arbitrary outcome classifications, EBM-COMET uses labels from a taxonomy recently published to standardise outcome classifications. Leveraging EBM-COMET and prior datasets, the thesis provides an in-depth analysis of state of the art contextual language models (CLMs) in terms of their capabilities and potential limitations in encoding and retrieving outcomes from clinical text. One main conclusion of this analysis is a consensus on which CLMs (BioBERT, SciBERT) are better suited to accurately detect mentions of outcomes in clinical text.

The thesis proposes a joint learning strategy that uses both word-level and sentence-level information to *simultaneously* perform outcome span detection and outcome type classification, both of which were previously performed separately. Experimental results on several benchmark datasets for health outcome detection show that my proposed joint learning method consistently outperforms decoupled methods. The thesis also proposes a position-based prompting strategy that queries language models to automatically generate health outcomes. It uses a position-attention mechanism to capture positional information of each word in a prompt relative to the mask to be filled, hence avoiding the need to re-construct prompts when the prompts' linguistic pattern changes. This approach demonstrates the ability of eliciting answers (in a case study on health outcome generation) to not only common prompt templates like Cloze and Prefix, but also rare ones too, such as Postfix and Mixed patterns whose masks are respectively at the start and in multiple random places of the prompt. More so, using various biomedical PLMs, the approach consistently outperforms a baseline in which the default PLMs representation is used to predict masked tokens.

The work in this thesis echoes the power and effectiveness of fine-tuning PLMs for domain-specific tasks such as health outcome detection and others embodied within biomedical text mining. I however recognize that it would be interesting future work to explore pre-training LMs before fine-tuning them for health outcome detection tasks as proposed under the future work section in the conclusion.

In summary, this thesis has demonstrated the potential Artificial Intelligence (AI), particularly NLP has in transforming (for the better) the way healthcare is delivered. The various propositions it makes and implements are timely interventions that can optimally increase the speed of analysis of clinical text for evidence based healthcare practices.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ABSA Aspect Based Sentiment Analysis

ACL Association for Computational Linguistics

BERT Bidirectional Encoder Representations from Transformers

BiLSTM Bidirectional Long short-term memory

BioNLP Biomedical Natural Language Processing

CLM Contextualised Language Model

CNN Convolutional Neural Network

COMET Core Outcome Measures in Effectiveness Trials

CRF Conditional Random Field

CRS Contextualised Representations

DDI Drug-Drug Interactions

DNEN Disease Named Entity Normalisation

DNER Disease Named Entity Recognition

DS Distant Supervision

EBM Evidence Based Medicine

EJBI European Journal for Biomedical Informatics

ELMo Embeddings from Language Models

EBM NLP Evidence Based Medicine-Natural Language Processing

EM Exact Match

EMNLP Empirical Methods in Natural Language Processing

ERNIE Enhanced Language Representation with Informative Entities

GloVe Global Vectors

IJCAI International Joint Conferences on Artificial Intelligence

KB Knowledge base

LCAM Label Context-aware Attention Model

LM Language Model

LM-as-KB Language models as knowledge bases

LSTM Long Short Term Memory

M-LP Multi-layer perceptron

ML Machine Learning

MLM Masked Language Modelling

MLP Multi-label Prediction

MNEN Medical Named Entity Normalisation

MNER Medical Named Entity Recognition

MT Machine Translation

MTL Multi-task learning

Mallet Machine Learning for Language Toolkit

MeSH Medical subject Headings

NB Naïve Bayes

NER Named Entity Recognition

NLI Natural Language Inference

NLP Natural Language processing

NLTK Natural Language Processing Toolkit

NN Neural Network

OC Outcome Type Classification

OD Outcome Detection

OSD Outcome Span Detection

PBL Prompt based learning

PBP Position-based Prompting

PICO **P**atients (P), **I**nterventions (I), **C**omparators (C) and **O**utcomes (O)

PLM Pretrained Language Model

PM Partial Match

PMI Pointwise Mutual Information

POS Part of Speech

QA Question Answering

RCT Randomised Clinical Trial

RCTs Randomised Clinical Trial abstracts

RE Relation Extraction

RNN Recurrent Neural Networks

RoBERTa **R**obustly **o**ptimised **B**ERT **a**pproach

SLC Sentence-Level Classification

SF Slot Filling

SOTA state-of-the-art

SVM Support Vector Machines

TL Transfer Learning

TLC Token-Level Classification

Tf-Idf Term frequency–Inverse document frequency

UMLS Unified Medical Language System

1 INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

Whatever the end goal is, skimming through a sizeable portion of text to understand and pinpoint relevant information is an extremely difficult task. It is laborious in nature, prone to error and subject to various interpretations which risk corrupting the actual communication in the text. This predicament is not unique to a single job, however, the ruthlessness of its damaging ripple effects can be more severe for some jobs. One such job in which a lot of unbearable damage can arise is health or patient care, for instance, prescription of wrong medication or imprecise diagnosis and prognosis resulting from inadequate search through clinical records can lead to untimely death [38, 132].

The need for effective and efficient searching through medical literature has never been more apparent than it is today. Hundreds if not thousands of clinical articles, reports and studies are published every single day in digital archives [121, 161]. To put it simply, medical literature is in overabundance. In fact, clinicians and researchers have acknowledged the difficulty in dealing with this sheer volume of literature [159, 206] in their day to day work. The challenge this presents is that, delivery of optimal health care is compromised when clinicians are unable to quickly access knowledge necessary in making informed clinical decisions.

The need for effective and efficient searching through medical literature has never been more apparent than it is today

Fortunately, computational methods and in particular Natural Language processing (NLP) have intervened and developed biomedical text mining tools to provide more expeditious literature searching and efficient clinical fact retrieval [110, 125]. For example given an article such as in Figure 1, to infer any piece of evidence, NLP would automatically process the article and extract the required evidence, whereas on the other hand, clinicians would need to manually read and interpret several segments in the article before deducing any evidence which can at times turn out to be false too as Figure 1 shows. BioNLP which connotes Biomedical Natural Language Processing (BioNLP) research has recently emerged under the umbrella field of NLP to primarily enable exploration and experimentation of text mining research methods applicable to literature from biomedical, clinical and generally health disciplines. To their credit, the NLP research community has achieved relative success in automating several BioNLP tasks such as chemical and disease relation extraction [127], chemical and drug recognition [67], gene identification [193],

extraction of Drug-Drug Interactions (DDI) [76, 188], protein recognition [15], de-identification or anonymisation of patient information [157] and extraction of biomedical evidence from clinical trials [107].

This thesis fits into the body of work that automates the extraction of biomedical entities from clinical text. Specifically it focuses on the extraction of health outcomes from Randomised Clinical Trial abstracts (RCTs) which are often made publicly available in digital repositories such as PubMed[1], Clinical Trials Registry[2] and Core Outcome Measures in Effectiveness Trials (COMET) Initiative.[3] Whereas some authors refer to them as variables monitored during clinical trials to assess the impact of studied interventions [52, 101, 114], the widely acknowledged definition of a health outcome is "a measurement or an observation used to capture and assess the effect of treatment such as assessment of side effects (risk) or effectiveness (benefits) [225]. In summary, they are often thought of as biomedical evidence [26, 52, 53]. For purposes of illuminating the biomedical evidence (health outcomes) extraction, the following section is used to broadly unveil the task of Outcome Detection (OD), its applications, challenges and profound relevance.

An outcome is "a measurement or an observation used to capture and assess the effect of treatment such as assessment of side effects (risk) or effectiveness (benefits)

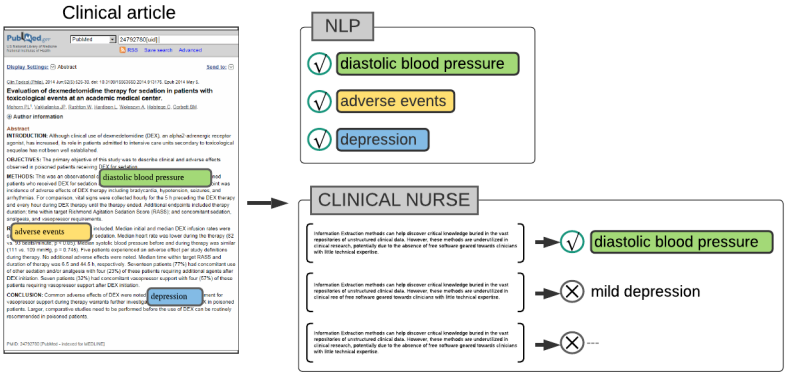


Figure 1: Given a clinical report/article/abstract, NLP systems (in contrast to manual approaches) are capable of more efficient and effective detection and extraction of clinically relevant information.

1.2 OUTCOME DETECTION

OD is subsumed by a bigger research paradigm called Evidence Based Medicine (EBM). EBM enforces healthcare decision making through the explicit and judicious use of current best evidence [181]. In practice, EBM researchers predominantly use a framework entitled **PICO** as a basis of formulating clinical questions to facilitate searching through biomedical literature. **P**atients (P), **I**nterventions (I), **C**omparators (C)

1 https://pubmed.ncbi.nlm.nih.gov/
2 https://clinicaltrials.gov/
3 https://www.comet-initiative.org/Studies

and **O**utcomes (O) (PICO) are essentially the critical elements of focus for clinicians when searching for relevant evidence i.e. **P**atients (or Population) stands for the patient/s of interest, their problem/s and demographics such as age, gender or ethnicity, **I**nterventions stands for the treatment/s or therapy being considered, **C**omparators (or Control) stands for a comparison treatment/s measured against the Intervention and the **O**utcomes stands for the observed or monitored evidence of effect during clinical trials [181]. Collectively considering all four elements significantly contributes towards various key health-care delivery indicators such as identification of evidence of the effectiveness of a certain treatment or diagnosis, strategies to evaluate quality of studies and mechanisms implemented in healthcare [88].

Despite not receiving attention equitable to other BioNLP research areas (such as disease, drug, chemical and gene recognition), the potential benefits of using readily available sources of clinical information has not completely gone unnoticed in EBM research. Of particular importance in EBM, is the identification of information about outcomes measured on patients [52], for example, blood pressure, fatigue, headache, pain etc. The ability to automatically detect outcomes (or outcome phrases) contained within clinical narrative text serves to maximise the potential of such sources. For example, GP letters or free text fields recorded within electronic health records, may often contain valuable clinical information which is not readily accessible or analysable without manual or automated extraction of relevant outcome. Similarly, automated identification of outcomes mentioned in trial registry entries or trial publications could help to facilitate systematic review processes by speeding up outcome data extraction. More so, the benefits of automated outcome recognition are increased further if it extends to categorisation of outcomes within a relevant classification system such as the taxonomy proposed in [55]. While it speeds up access to the best available evidence in context of patients' individual conditions [52], automated identification of outcomes is also a cost effective method that aids delivery of optimal patient care [26].

To minimise compromised patient care, clinicians need just-in-time access to the best available evidence in context of the patients' individual conditions

1.2.1 Challenges in Outcome Detection

Similar to other clinically relevant entities, outcomes are primarily mentioned in scientific publications which are disseminated as unstructured text. This introduces two challenges, the first being identifying publications that best describe clinical trial outcomes from a plethora of publications and second being, extracting target phrases that correspond to outcomes from each relevant study identified [52]. Fortunately, outcomes are predominantly reported in RCTs, hence, narrowing down the search space of relevant studies in the OD process.

However, multiple researchers indicate that, there are a lot of inconsistencies in outcome reporting resulting from the variability in how outcomes are defined and measured across several clinical trials [55, 88, 225]. Furthermore, Williamson et al. [225] note that, there is a significant degree of outcome reporting bias where studies with statistically significant results on outcomes measured are more likely to be published than others.

The above problems result from or are further exacerbated by the lack of a uniform classification system that absorbs all outcomes that patients regard as most important or relevant [55]. Despite the initiatives to standardise outcomes [210], there is hardly any dictionary nor vocabulary for reference of all outcome terminology unlike other clinical entities such as diseases which have repositories of terminologies to support their extraction. All together, these challenges make the task of detecting and extracting outcomes from biomedical literature burdensome. Moreover, the subject has attracted less attention than it might from the BioNLP community. At this point, it is predictable that OD is a low-resourced task with limited publicly available annotated corpora and limited benchmarking tools and applications that can be applied across several corpora. Nonetheless, the BioNLP community has taken some positive strides and proposed different methods to alleviate some of the aforementioned challenges as the next chapter discusses.

Outcome reporting is inconsistent across several RCTs because there is no consensus on how outcomes should be reported. Limited annotated corpora has led to less attention form the BioNLP community

1.2.2 Outcome Detection in NLP

Earlier work on NLP for OD cast it as a text classification task where the goal was to classify sentences in RCTs as outcome-statements (task (1) in Table 1), which indicated that the sentences summarised the consequences of an intervention [23, 53]. Using classifiers like Naïve Bayes (NB) [84], Support Vector Machines (SVM) [73], Multi-layer per-

	Task	Output
(1)	**Outcome statement classification** Demner-Fushman et al. [52]-2006, (Boudin, Nie, and Dawes [23], Boudin et al. [22])-2010, (Kim et al. [107], Huang et al. [85])-2011, Wallace et al. [214]-2016	- outcome statement
(2)	**Outcome Span Detection** OSD) Nye et al. [161]-2018, (Kang, Zou, and Weng [101] Brockmeier et al. [26])-2019	- wheezing - shortness of breath
(3)	OC) Nye et al. [161]-2018, Abaho et al. [1]-2019	- Physiological outcome

Table 1: The evolution of OD tasks chronologically ordered from what it was before, to what it was at the point of commencement of the work covered in this thesis.

ceptron (M-LP) [30, 85], these works built classification models to predict whether a sentence contains an outcome or not. Later on, the task was re-modelled as either (a) OSD, a *sequence labelling task* where the goal is to detect which text spans in a Randomised Clinical Trial (RCT) abstract describe health outcomes (task (2) in Table 1) [26, 161], or (b) OC, a *classification task* where the goal is to classify a text into a predefined set of outcome types or categories depending on an outcome that is mentioned somewhere in that text (task (3) in Table 1) [1, 161].

Despite being restrained by shortage of expertly labeled datasets, few attempts to create EBM-oriented datasets to support OD have been made. Wallace et al. [214] use Distant Supervision (DS) to annotate sentences in clinical trial articles with PICO elements. Demner-Fushman et al. [52] use an experienced nurse and a medical student to annotate outcomes by identifying and labelling sentences that best summarise the consequence of an intervention. Similarly, other attempts have precisely segmented PubMed abstracts into sentences that they label one of P, I and O (I and C are collapsed into I) to respectively imply Patients, Interventions and Outcomes [95, 107]. Since annotation of the above datasets is tailored for Sentence-Level Classification (SLC), it becomes difficult to use them for individual PIO elements extraction tasks [26, 101] such as OD. This difficulty henceforth propelled works such as Nye et al. [161] to annotate granular P, I, O information within RCTs using a mixture of crowd workers (non-experts) and expert workers, thus, producing the recently released EBM-NLP corpus.

The early OD datasets could only support sentence level classification of outcomes, however this work builds on recent EBM-NLP dataset with token level outcome annotations to further the task of OD.

Carefully reviewing the above as well as in several sections of the following thesis chapters, it is rather noticeable and evident that OD and Evidence Based Medicine-Natural Language Processing (EBM NLP) have a dearth of expertly labelled datasets. Additionally, I am aware of a well documented challenge (earlier mentioned in Section 1.2.1) of the absence of a consensus on how clinical trial outcomes should be classified [45, 53, 55], which has in turn detracted attention of computer scientists from the task. These challenges therein motivate the work covered in Chapter 3 of this thesis.

In spite of the scarcity of publicly available expertly annotated corpora highlighted in the previous paragraph, the rapid advancement in NLP techniques has accelerated EBM NLP. Several works have adopted the impressive artificial neural network architectures such as Bidirectional Long short-term memory (BiLSTM) [77] to enhance automated PICO elements extraction. Using a Conditional Random Field (CRF) [116] as an output layer, the performance achieved in PICO elements extraction (cast as a Named entity Recognition task) has further improved using the EBM-NLP corpus [95, 101, 161]. Of late, a small number of authors have used Transfer Learning (TL) as a conduit to adopt Contextualised Language Model (CLM)s such as SciBERT [18] and ClinicalBERT [8, 113] to achieve PICO extraction. CLMs

such as the aforementioned, account for information about surrounding words when processing a single word, and because of this, they have achieved unparalleled success in traditional BioNLP tasks including disease, gene, drug and chemical name recognition [121, 211, 241]. However, the same cannot be emphatically stated for outcomes. This then, inspired the attempt to enhance current state-of-the-art (SOTA) performance in OD made in Chapter 4 and Chapter 5.

1.3 LANGUAGE MODELS AS HEALTH OUTCOME KNOWLEDGE BASES

How robustly is information relating to health outcomes stored in Language Models

Language Model (LM)s have overwhelmingly thrived in inferential statistics, proving an arguably indisputable mark in the interpretation, decoding and disambiguation of written language [54, 136, 167]. This success has instigated the NLP research community to further explore LMs by asking questions such as, how knowledgeable are these LMs, in other words, what do they know and how much of it do they know? To obtain answers, a whole new research paradigm deemed Language models as knowledge bases (LM-as-KB) has emerged in which, researchers probe LMs for factual knowledge that presumably was learned during their training and as such is transferable and associable [27, 94, 128, 170, 190]. For example, if an LM learns that "doctors" *treat* "patients", will it always be able to associate doctors with the treatment of patients since the statement might not always be explicitly stated, and further more, will it be able to distinguish this "doctor-patient" relationship from various other relationships that doctors or patients have with other entities such as medicine. In essence, this task queries LMs for stored retrievable world knowledge and the extent to which they have grounding in perception of facts expressed in language [75].

Multiple works on this subject have emerged making use of "fill-in-the-blank" statements and thereby, tasking LMs to accordingly predict the expected information in the blanks. These "fill-in-the-blank" statements (often referred to as prompts) have severally been used in fine-tuning and querying a Pretrained Language Model (PLM) for relational knowledge between different entities. While there have been multiple works querying for general-domain relational knowledge such as where people live and work, where and when people were born or died [94, 170], little has been done in terms of querying LMs for more complicated and domain-specific relational knowledge. Using triples directly extracted from established biomedical Knowledge base (KB)s like the Unified Medical Language System (UMLS)[4] and Comparative Toxicogenomics Database (CTD)[5], Sung et al. [200] recently inspected the potential utility of LMs as biomedical KBs. The challenge with this is, not only are the biomedical entities explored

4 https://www.nlm.nih.gov/research/umls/index.html
5 http://ctdbase.org/

eligible for being cast into (*subject-relation-object*) triples e.g. *headache-symptomOf-Pituicytoma*, there is no existing KB with this relational knowledge structure for health outcomes [55].

To investigate the utility of LMs as health outcome knowledge bases, this thesis therefore designed a novel task named outcome generation to probe several biomedical LMs by querying them to (1) recall outcome information encountered during training and (2) generate outcomes using out of scope prompts or prompts never encountered during training. This investigation is covered in Chapter 6 of the thesis.

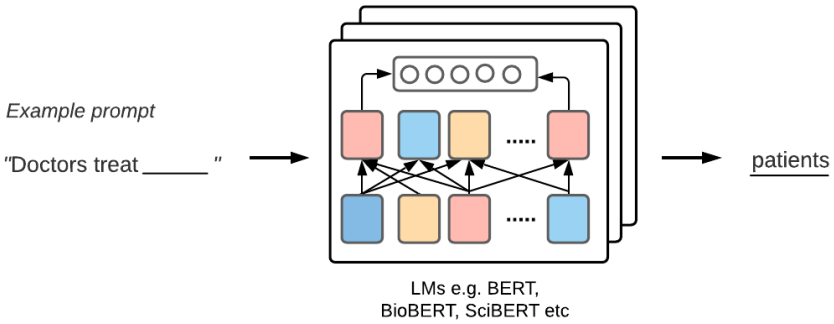


Figure 2: Querying a language model (LM) as a knowledge base for factual knowledge.

1.4 RESEARCH AIM AND OBJECTIVES

In light of the preceding discussion on the OD task, its evolution in NLP and the identified gaps motivating this work, the chief aim of this thesis is:

> *To enhance and advance automated extraction of health outcomes in evidence-based medicine/practice using natural language processing methods and the abundant literature in the biomedical domain. The main focus is on the explicit identification of individual outcome spans and classification of outcomes into outcome types to enhance the search and retrieval of evidence.*

To further unpack this aim, the specific objectives and targets that are worked towards herein include the following,

. . . Enhance and advance the health outcome identification and classification.

1. To evaluate and improve the reliability of current outcome annotations, with a keen emphasis on weakly labelled datasets. Subsidiary to this objective, is a list of more specific targets as outlined below,
    1.1. To tackle flaws in outcome annotations in datasets currently supporting OD.

1.2. To denoise weakly labelled crowd sourced outcome annotations.

1.3. To align outcome annotations to standardised outcome classification systems.

2. To assess and advance the OD task to new SOTA performance on benchmark datasets. Specifically, this objective is further narrowed down to these targets,

2.1. To build and fine-tune custom LM architectures that are superior, more competent and re-usable than current OD approaches.

2.2. To obtain a consensus on which PLMs are best suited for the OD task.

2.3. To publicly avail PLMs that can be fine-tuned for the OD task.

This thesis seeks to assess and advance the task of OD to new SOTA performance on benchmark datasets.

3. To probe for methods that can explicitly identify categorical outcome knowledge specific to individual outcome spans from RCTs.

4. To empirically verify and justify the evaluation performances obtained by methods proposed in objectives (2) and (3).

1.5 CONTRIBUTIONS

Several investigations were conducted in view of the aim, objectives and targets stated above. As a result, a number of milestones (included below) are arrived at, which subsequently served as a sufficient pretext for the preparation of this thesis.

- A hybrid strategy to denoise flawed outcome annotations. This re-usable strategy involves combining Part-Of-Speech tagging and Rule-based chunking to automatically identify and correct incorrect demarcations of outcome spans. This work is presented in Section 3.2 and published at the International Joint Conferences on Artificial Intelligence (IJCAI) 2019 KDH workshop [1]. Resources used including the code and dataset are publicly availed.[6]

- A flexible, re-usable label alignment approach that extracts pseudo parallel annotations from comparable datasets. This approach is aimed to primarily denoise outcome label annotations by aligning them to more recent standard outcome classification systems that were never used in their annotation in the first place. The approach is proposed within work published at the Empirical Methods in Natural Language Processing (EMNLP)

6 https://github.com/MichealAbaho/pico-outcome-prediction

2021 conference [3] and presented in Section 3.4. The code is published.[7]

- In addition to building a custom OD classification model that consumes contextual embeddings, the thesis provides a comparative assessment of biomedical CLMs in the OD tasks. This analysis is extensively covered in Chapter 4 and was published in European Journal for Biomedical Informatics (EJBI) 2021 [2]. Both code and dataset are provided.[8]

- This work constructed a new dataset by collecting a set of RCTs from PubMed and expertly annotating outcomes in the RCTs. It additionally presented empirical evidence of the benefit of fine-tuning several biomedical PLMs for OD using the dataset. The dataset construction details are included in Section 3.3 and published.[9]

- A joint learning strategy that simultaneously achieves health outcomes span detection and health outcome type classification without compromising the performance of either one of the tasks. This work is proposed and published in EMNLP 2021 [3] and presented in Chapter 5. Code and datasets are publicly accessible.[10]

- A novel position-attention prompting framework to probe LMs for knowledge relevant to health outcomes. This prompting framework is expounded on in Chapter 6 and has been published in the Association for Computational Linguistics (ACL) 2022 BioNLP Workshop. Code used is publicly accessible.[11]

1.5.1 Publications

[1] Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. "Correcting crowdsourced annotations to improve detection of outcome types in evidence based medicine." In: *CEUR Workshop Proceedings*. Vol. 2429. 2019, pp. 1–5. URL: http://ceur-ws.org/Vol-2429/paper1.pdf.

[2] Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. "Assessment of contextualised representations in detecting outcome phrases in clinical trials." In: *European Journal of Biomedical Informatics* 17.9 (Aug. 2021). URL: https://arxiv.org/pdf/2203.03547.pdf.

7 https://github.com/MichealAbaho/Label-document-Alignment
8 https://github.com/LivNLP/ODP-tagger
9 https://github.com/LivNLP/ODP-tagger/tree/master/EBM-COMET
10 https://github.com/MichealAbaho/Label-Context-Aware-Attention-Model
11 https://github.com/MichealAbaho/outcome_generation

[3] Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. "Detect and Classify – Joint Span Detection and Classification for Health Outcomes." In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 8709–8721. DOI: 10.18653/v1/2021.emnlp-main.686. URL: https://aclanthology.org/2021.emnlp-main.686.

[4] Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. "Position-based Prompting for Health Outcome Generation." In: *Proc. of The 21st BioNLP workshop associated with the ACL SIGBIOMED special interest group*. 2022. URL: https://arxiv.org/abs/2204.03489.

1.6 THESIS ORGANISATION AND SUMMARY

Beyond this point, this thesis contains 6 other chapters which are organised as follows,

CHAPTER 2 - RELATED WORK: Presents a variety of works that are similar to this thesis' contents in mainly two ways, either conceptually in terms of application of NLP methods in mining clinically salient information from biomedical literature or in terms of biomedical evidence extraction with an inclination towards health outcomes.

CHAPTER 3 - REFINEMENT AN ANNOTATION OF OUTCOME DATA: Presents and describes the different methods and propositions implemented to overcome the scarcity of reliable resources to support biomedical evidence extraction, and in particular OD. Besides proposing two scalable and flexible denoising methods (outcome span denoiser and outcome label denoiser), the chapter presents an expertly labelled dataset to support the OD task.

CHAPTER 4 - ASSESSMENT OF CONTEXTUALISED REPRESENTATIONS IN DETECTING OUTCOMES: Presents two TL adaptation mechanisms that leverage several existing biomedical CLMs in building models that produce SOTA performance in the OD task.

CHAPTER 5 - JOINT SPAN DETECTION AND CLASSIFICATION FOR HEALTH OUTCOMES: Presents and describes a joint learning strategy that maximizes the word- and sentence-level information (in RCTs) to simultaneously achieve outcome span detection and outcome type classification.

CHAPTER 6 - POSITION-BASED PROMPTING FOR HEALTH OUTCOME GENERATION: Proposes a novel prompting mechanism to probe LMs for factual health outcome related information.

CHAPTER 7 - CONCLUSION: Summarises the work covered in the thesis with a summary of the highlights, a discussion of the challenges and limitations encountered, a discussion of the applicability of the different methods proposed and finally, possible future work beyond this thesis.

2

RELATED WORK

2.1 INTRODUCTION

The unprecedented volume of biomedical research articles published every day necessitates continually overhauling biomedical literature searching methods. Courtesy of the acceleration and advancement of computational linguistics research initiatives such as NLP as well as the increase in computational power, a multitude of methods, techniques and datasets are increasingly published to enhance biomedical information searching and retrieval. Many of these resources served as a building block to motivate the work that was undertaken and presented in this thesis. Because of the profound role PLMs play in the methods proposed by this thesis, this chapter begins by providing a brief background of TL and its recent successes in BioNLP in Section 2.2. Following a preamble with a brief introduction to EBM NLP, the chapter discusses and reviews the main problem of identifying individual outcome spans and classifying outcomes into outcome types that this thesis largely tackles as mentioned in the preceding Chapter 1. To motivate the work on denoising crowdsourced annotations in Section 3.2, this chapter investigates published datasets that are constructed to support OD (Section 2.3.2), and as well review prior work on noise reduction in weakly labelled BioNLP datasets in Section 2.4. Subsequently, the chapter extensively details prior approaches undertaken to achieve OD which can be categorised into two branches, sentence- and token- level classification of outcomes. Towards its end, the chapter reviews attempts that have been made in joint modelling strategies to enhance information extraction in the clinical domain within Section 2.5. Furthermore, the chapter discusses efforts made in treating LMs as KBs in Section 2.6. Last but not least, the chapter provides brief descriptions and backgrounds of various other BioNLP tasks in Section 2.7, that may in one or more ways relate to OD.

2.2 TRANSFER LEARNING (TL)

TL is an ML approach that enables usage of a model to achieve a task that it was not initially built and trained for.

TL is a Machine Learning (ML) approach that enables usage of a model to achieve a task that it was not initially built and trained for [199]. Usually, the assumption is that, train and test data for a specific task exists, however, this is never the case. TL therefore allows learning across different tasks. The term pre-trained often used in TL approaches and tasks implies that a model was previously trained on a task different from the target task it is currently being used for.

The most popular PLMs adopted for TL include the Uni-directional or auto-regressive LMs; which given a sequence $s = \{w_t\}_{t=1}^{|s|}$, model $P(w_t|w_{<t} : \theta)$, the probability of a target word w_t given the previously seen words or context $w_{<t}$ such as [GPT-3; 27], [Text-to-Text Transfer Transformer T5; 174] and [XLNet; 235]. Some popular variants of auto-regressive LMs are Left-to-Right (L2R) LMs such as Encoder-Decoder models [40], which use two separate neural architectures to model $P(y_{<N_{dec}}|x_{<N_{enc}})$, with the first architecture encoding an input sequence $x_1, \ldots, x_{N_{enc}}$ and the second architecture decoding an output sequence $y_1, \ldots, y_{N_{dec}}$ conditioned on the input sequence representation. N_{enc} and N_{dec} are the encoder input and decoder output sequence lengths respectively.

The other popular PLMs are the Bidirectional LMs; which given s defined in above paragraph, model $P(w_t|w_{1\leqslant t-1,t+1\geqslant n} : \theta)$, the probability of a word given the surrounding context such as [Bidirectional Encoder Representations from Transformers (BERT); 54], [**R**obustly **o**ptimised **BERT** **a**pproach (RoBERTa); 136] and [Embeddings from Language Models (ELMo); 167]. Many Bidirectional LMs use Masked Language Modelling (MLM), a paradigm which masks pieces of an input (words and or sub-words depending on the tokenization algorithm [228]) and then trains the model to predict the masked tokens given the surrounding context. Generally Both Uni- and Bi- directional LMs are hinged upon the Transformer architecture (proposed by google research team [213]) characterised by three important aspects that make them distinct from earlier Neural Network (NN) architectures like the Recurrent Neural Networks (RNN), Long Short Term Memory (LSTM) and Convolutional Neural Network (CNN). These aspects include non-sequential processing, i.e sentences are processed as a whole rather than word by word, self-attention mechanism, which allows them compute similarity scores between the words in a sentence, hence, they are able to consider the impact every other word in a sentence has when predicting a particular word and, finally positional embeddings which encode information related to a specific position of a word in a sentence [213].

Uni-directional and Bidirectional Language Modelling are two common and pre-training methods responsible for the success in TL

CLMs such as the above mentioned Uni- and Bi- directional LMs have significantly outperformed context-independent embeddings such as word2vec [149] and Global Vectors (GloVe) [166] in various TL downstream NLP tasks such as Question Answering (QA), a task to automate the answering of questions [175], Textual entailment, a task of determining whether a given "hypothesis" is true given a "premise" [24], Named Entity Recognition (NER), a task to extract different types (such as names, location etc) of entities mentioned in text [207] etc.

The success of general-domain TL propelled the emergence of domain-specific TL and in particular, domains such as biomedicine through BioNLP have remarkably advanced of late. The notion of general-domain is used to denote a distribution over a language characterising a

diverse set of topics, whereas domain-specific is used to denote a distribution over a language characterising a single topic. BERT variants such as SciBERT [18], ClinicalBERT [10] and PubMedBERT [70] yielded performance improvements in the Biomedical NER tasks on the BC5DR dataset [56, 127], text-classification tasks like Relation Extraction (RE) on the ChemProt dataset [115] and on extraction of PICO elements. Despite being pre-trained on English biomedical text, BioBERT [121] outperformed the generic BERT model (pre-trained on Spanish biomedical text) in Pharma-CoNER, a multi-classification task for detecting mentions of chemical names and drugs from Spanish biomedical text [199]. Recently, Jin et al. [97] discovered that, in comparison to BioBERT, BioELMo (Biomedical ELMo) better clustered entities of the same type such as, an acronym having multiple meanings or a homonym. For example, unlike BioBERT, BioELMo clearly differentiated between ER referring to "Estrogen Receptor" and ER referring to "Emergency Room" in their work.

Most notably, TL is the fundamental aspect that links up various works covered in the majority of this thesis' chapters. Adoption of PLMs for domain-specific tasks in a TL setup has not just eliminated the need to train models from scratch, but it has also led to performance improvements in tasks that are defined and reviewed in the subsequent sections of this chapter.

2.3 EVIDENCE BASED MEDICINE NATURAL LANGUAGE PROCESSING (EBMNLP)

EBM NLP is term used to refer to the adoption and application of NLP techniques to extract evidence of effective interventions from biomedical literature. Clinically, evidence is often arrived at through manually searching for answers (within clinical text documents) to PICO formatted questions. From a BioNLP standpoint, researchers have invented different approaches of automatically adducing this type of evidence, i.e. evidence that entails or stipulates all or some of the PICO elements. EBM NLP work has largely concentrated on the task of automatic recognition of PICO elements making use of neural architectures like the M-LPs, RNNs, LSTMs [26, 96, 101] and CLMs [18, 70] in the recent past. Of late, EBM NLP has diversified in terms of approaches, with some authors designing Natural Language Inference (NLI) tasks to deduce evidence from biomedical literature. Lehman et al. [122] build a model to predict whether a clinical article (treated as a premise) suggests that an Intervention specified in a prompt (treated as a hypothesis) led to a significant increase, decrease or no difference to a health outcome. While most works often detect PICO elements all together, this work pays keen and specific attention to detection and extraction of health outcomes as earlier motivated in Chapter 1.

EBM NLP involves the adoption of NLP techniques to extract evidence of effective interventions from biomedical literature

2.3.1 Outcome Detection (OD)

OD is systematic in nature, with some authors describing it as two-fold in nature, where first, a minimal amount of text sufficient enough to understand implications of health outcomes is determined, and secondly, the text units that describe health outcomes are identified [52]. Automatic execution of both or any of these tasks is rapidly becoming a norm given the recent surge in text-mining tools as highlighted in Chapter 1. The next section seeks to uncover the developments in terms of datasets facilitating automatic OD as well as categorically discuss the two classification approaches used in automatic OD.

2.3.2 Datasets

Prior work on OD and EBM NLP has been limited by the scarcity of publicly available corpora for training and evaluation [161, 197]. According to Dodd et al. [55], this gap mainly emanates from the lack of standard classification systems for not just outcomes, but PICO elements as earlier mentioned in the challenges described under Section 1.2.1. Additionally, I observe that researchers struggle to recruit expert annotators and they heavily rely on the structured nature of RCTs to prepare datasets for this task. Subsequently, the annotation guidelines for EBM datasets summarised in Table 2, have varied from one study to another as noticed in previous construction efforts.

Prior work on OD and EBM NLP has been limited by scarcity of publicly available corpora for training and evaluation.

Focused on 3 conditions (Rheumatoid arthritis, migraines and breast cancer), Demner-Fushman et al. [52] employed a team of 2 experienced clinical nurses, a medical student and a PhD to identify sentences containing health outcomes within 633 Medline articles [155]. Their annotation scheme consisted of Medical subject Headings (MeSH)[1] definitions of 7 elements that were to be used in tagging sentences. These were "Background", "Population", "Intervention", "Statistics", "Outcome", "Supposition" and "Other". Their understanding and hence guiding definition of an outcome was "a sentence that best summarizes the consequence of an intervention".

Kim et al. [107] slightly edited Demner-Fushman et al. [52]'s annotation scheme by excluding "Statistics" and "Supposition" categories and instead introduced "Study Design" category to annotate a bigger number of Medline articles (1000). Kim et al. [107] further advanced the selection of RCTs to include in the dataset, by using queries provided by two institutions, the Global Evidence Mapping Initiative (GEM)[2] and the Agency for Healthcare Research and Quality (AHRQ)[3] to retrieve RCTs linked to traumatic brain and spinal cord injury.

1 https://www.nlm.nih.gov/mesh/meshhome.html
2 http://www.evidencemap.org/
3 http://www.ahrq.gov/

Demner-Fushman et al. [52] and Kim et al. [107] annotation process assumed that the outcomes and or other PICO elements can be mentioned anywhere in the text. Several works later gave precedence to the structured nature of the RCTs in identifying and thereby annotating PICO elements. Boudin et al. [22], Boudin, Nie, and Dawes [23], Huang et al. [84, 85], and Jin and Szolovits [96] all follow the distinct section headings in the abstracts to annotate sentences such that, all sentences under a section heading PARTICIPANTS were labelled "P", sentences under a section heading INTERVENTIONS were labelled "I", sentences under a section heading COMPARATORS were labelled "C" and finally sentences under the section heading OUTCOMES were labelled "O". Some key word section headings might not appear verbatim to the headings of interest, and therefore synonymous phrases were considered such as PATIENTS/POPULATION/-SUBJECTS if PARTICIPANTS is not found, MEASUREMENTS/MEASURED OUTCOME/CLINICAL OUTCOME if OUTCOMES is not found.

To annotate even larger corpora, Wallace et al. [214] adopt an approach that distantly supervises annotation of 12,808 structured and semi-structured RCTs from the Cochrane Database of Systematic Reviews (CDSR) [195]. The lay annotators weakly labelled article sentences as negative or positive by finding at least 4 overlapping tokens across the article sentences and free-text summaries descriptive of PICO elements in CDSR.

Most recently, researchers are adopting annotation tools to enhance preparation of datasets to facilitate EBM. Kang, Zou, and Weng [101] employed a medical professional and an informatics researcher to annotate 170 Medline RCTs using BRAT, a web based collaborative annotation tool [196]. Each abstract is initially classified into 5 common clinical question types, Treatment, Prevention, Diagnosis, Prognosis and Etiology. Thereafter, two attributes including a Qualifier (qualitative description e.g. "different", "similar", "higher") and a Measure (quantitative description e.g. "138+/- 13mg daily") were used in identifying PICO elements.

The challenge with earlier datasets supporting OD and EBM NLP, is that, only sentence-level annotation was conducted and entity level annotation of outcomes or other PICO elements was neglected

The challenge with all of the above datasets is that, only sentence-level annotation was conducted and entity level annotation of outcomes or other PICO elements was neglected. It therefore, becomes difficult to use them for tasks that require extraction of individual PICO elements [26, 101] such as OSD. To address this issue, Nye et al. [161] published EBM-NLP, a dataset in which ca. 5,000 clinical trial abstracts were annotated with PICO elements by a mixture of lay and expert annotators. The corpus has two versions, (1) the "**starting spans**" in which text spans are annotated with the literal "PIO" labels (I and C merged into I) and (2) the "**hierarchical labels**" in which the annotated outcome "PIO" spans were annotated with more specific labels aligned to MeSH terms, for instance the Outcome (O) spans are

annotated with more granular (specific) labels which include Physical, Pain, Mental, Mortality and Adverse effects.

The EBM-NLP dataset has however been discovered to have flawed outcome annotations [1] such as (1) statistical metrics and measurement tools annotated as part of clinical outcomes e.g."*mean arterial blood pressure*" instead of "*arterial blood-pressure*","*Quality of life Questionnaire*" instead of "*Quality of life*" and (2) Multiple outcomes annotated as a single outcome "*Systolic and Diastolic blood-pressure*" instead of "*Systolic blood-pressure*" and "*Diastolic blood-pressure*". Further more, similar to the earlier discussed datasets and summarised in Table 2, construction efforts of this granular detailed PICO dataset lacked a standard classification system to accurately inform the annotation process and instead opted for arbitrary labels such as those terms aligned to MeSH. To address these concerns, this work proposes a couple of denoising methods and leverages a recently released standardised outcome taxonomy [55] to expertly construct a dataset of health outcomes in Section 3.3.

2.3.3 Sentence level classification (SLC)

Biomedical Semantic analysis, Clinical semantic text similarity, Medical NLI and OD are a few of the SLC oriented domain-specific (Clinical) tasks.

NLP tasks setup to predict one or more predefined classes given a sequence of words in a sentence are often referred to as SLC tasks [24, 46, 109, 172]. Some authors interchangeably use the terms document and sentence, however both will almost always imply a sequence of tokens such as $(w_1, \ldots, w_n)$ that can belong to a larger body of text such as an article, a newspaper, a review, a journal paper etc. The vast amount of general-domain or domain-specific applications underpinned by SLC as applied in NLP is arguably unfathomable. Spam detection [144], customer review comprehension [172], theme/topic detection [177], sentiment analysis [164] are a few of the general-domain applications modelled as SLC tasks, whereas, Biomedical sentiment analysis [208], Clinical semantic text similarity [217] and Medical Natural Language Inference [179] are a few of the SLC-oriented domain-specific applications.

In the same vein, OD has previously been cast as a SLC task, in which a given abstract sentence is classified as an outcome statement (sentence summarising the consequences of an intervention) or not [23, 53]. Several other authors have classified abstract sentences into one of four labels, Participants (P), Interventions (I), Comparators (C) and Outcomes (O) [22, 96, 107]. Section 2.3.2 earlier introduced a list of datasets (and also summarised them in Table 2) that have supported SLC for OD. Below, I extensively discuss various methods and approaches implemented to achieve SLC for OD.

Dataset	Source of Abstracts & search strategy	Abstract type	Annotators	Inter-Annotator agreement	Task
Demner-Fushman [52] Total abstracts - 592	Medline Searched for topics (arthritis, rheumatoid, migraine, breast cancer, diabetes, Immunistation)	RCT Both Structured and Unstructured	Nurse Medical Student PhD	P - 0.75 I - 0.75 O - 0.75	Sentence classification
Kim et al [107] Total abstracts - 1000	Medline Queries from GEM and AHRQ	RCT Both Structured and Unstructured	Medical Student under supervision	P - 0.63 I - 0.61 O - 0.71	Sentence classification
Jin et al [96] Total abstracts - 489026	Medline English RCTs published	RCT Structured	Expertise not mentioned	N/A	Sentence classification
Boudin et al [23] Total abstracts - 50	Medline RCT publication type	RCT Structured	Expertise not mentioned	N/A	Sentence classification
Boudin et al [22] Total abstracts - 26000	Medline RCT publication type	RCT Structured	Medical professionals	N/A	Sentence classification
Huang et al [85] Total abstracts - 23472	Medline PubMed abstracts	RCT Structured	Non-expert	N/A	sentence classification
Huang et al [84] Total abstracts - 489026	Medline RCT publication type	RCT Structured	Non-expert	N/A	sentence classification
Wallace et al [214] Total abstracts - 12808	CDSR Clinical trials	Structured & Semi-structured	Non-expert	0.81	sentence classification
kang et al [101] Total abstracts - 170	Medline RCT publication type	RCT Unstructured	Medical professional Informatics Researcher	0.83	Token-level classification
Nye et al [161] Total abstracts - 5000	Medline RCTs with an emphasis on cardiovascular diseases, cancer and autism	RCT Unstructured	Non-expert Expert Medical students and doctors	P - 0.50 I - 0.59 O - 0.51	Token-level classification

Table 2: Summary of datasets supporting PICO detection at sentence- and token-level. Information about the source of the abstracts, and the search strategy used in selecting the abstracts retrieved for annotation, the Abstract type, the level of expertise of the annotators, the Cohen Kappa Inter-annotator agreement and the task the dataset was prepared for.

2.3.3.1 Conventional Machine Learning based Approaches

To classify sentences in RCTs into one of four PICO elements, earlier works modelled representations of hand-crafted features using conventional ML methods such as NB and SVM implemented in ML toolkits such as Machine Learning for Language Toolkit (Mallet)[4] and Natural Language Processing Toolkit (NLTK).[5]

Demner-Fushman et al. [52] select the top 3 ranked sentences of a list of sentences output as outcome-statements by an ensemble of classification algorithms. The stacked algorithms trained using Mallet included (1) a Rule based classifier, which relied on cue-phrases like "significantly greater", "adverse events" etc to estimate the likelihood of an outcome statement, (2) a NB classifier, which generated the

4 https://mimno.github.io/Mallet/
5 https://www.nltk.org/

probability of a sentence (treated a bag of words) being an outcome statement, (3) an n-gram-based classifier, which used uni-gram and bi-gram (a single or pairs of words appearing consecutively in a sentence) features selected on an information gain measure basis [234] to estimate the probability of an outcome statement, (4) a position classifier, which relied on the discourse structure of an abstract, i.e. which position within the abstracts is an outcome statement likely to be found, (5) a document length classifier, which returned a smoothed probability that a document of given length contains an outcome statement and finally (6) a semantic classifier, which generates the likelihood of a given sentence being an outcome statement on the basis that it contains UMLS concepts highly associated with outcomes.

Instead of cue-phrases, Kim et al. [107] use a set of four features including (1) bi-grams and Part of Speech (POS) tags, (2) Concept Unique Identifiers (CUIs) and their synonyms extracted from UMLS, (3) structural information such as headings of various sections in the abstract and (4) sequential information which included direct and indirect dependencies between sentences. These features are trained using a CRF to classify sentences into categories that did not just include P, I and O but also "Background" and "Study design". "Background" implied a sentence belonged to a section in the abstract that informs or preceded current study and "Study design" implied the type of study described in the abstract. Besides the semantic features (CUIs), all of the other features improved the performance of the classifier and they attributed this to the sparseness and ambiguity of the terms found during querying UMLS.

Earlier works heavily relied on hand-crafted features and the structural ordering of information in RCTs.

Huang et al. [84] also use structural information however differently from Demner-Fushman et al. [52] and Kim et al. [107]. They build two sets of NB classifiers, the first set including CF_P, CF_I and CF_O separately built by training on the first sentence following the sections in the abstracts that are respectively labelled Participants (P), Interventions (I) and Outcomes (O). The second set including CA_P, CA_I and CA_O is similar, however built by training on all sentences following the respective similarly predefined section headings as the first set. Overall, their study revealed that, the very first sentence under abstract sections headings particularly Participants, Interventions and Outcomes does not always contain information relevant to P, I and O elements, which many prior authors had assumed.

Boudin, Nie, and Dawes [23] reduce the reliance on many hand-crafted features in prior work by adopting a language modelling approach that models the probability of individual words for the PICO sentence retrieval task. After exploiting the positional distribution of PICO elements by dividing each abstract into 10 parts of equal length and marking them $[P1, \ldots, P10]$, they build two LMs, M_q representing a query q that is searching for one of P,I and O components and M_d representing a document d being classified. To score how relevant d

is to q, a Kullback-Leibler divergence (KL) score is computed using (1),

$$\begin{aligned} \text{score}(q, d) &= \sum_{w \in q} P(w|M_q) \cdot \log P(w|M_d) \\ &\propto -\text{KL}(M_q \parallel M_d) \\ P(w|M_s) &= \frac{\text{count}(w, s)}{|s|} \end{aligned} \tag{1}$$

where $s \in \{q, d\}$ and $\text{count}(w, s)$ is the number of times the word w occurs in s and $|s|$ is the length of document s. The LM $P(w|M_d)$ is extended to (2) such that it integrates the structure of the 10 parts i.e. a weighted linear interpolation factor γ_p is assigned to each $p \in [\text{TITLE}, P1, \ldots, P10]$,

$$P_1(w|d) \propto P(w|M_d) + \sum_{p \in d} \gamma_p \cdot P(w \in p|M_d) \tag{2}$$

and the LM $P(w|M_q)$ is extended to (3) to factor in presence of a PICO element in a query i.e. a weight δ_e is given to a query word belonging to the elements $e \in [P, I, C, O]$, $f(w, e) = 1$ if $w \in e$, 0 otherwise.

$$P_2(w|M_q) \propto P(w|M_q) + \sum_{e \in [P,I,C,O]} \delta_e \cdot f(w, e) \cdot P(w|M_q) \tag{3}$$

Recent research achieving SLC of sentences for PICO detection reduced reliance on hand-crafted features and instead adopted language models that model the probability of each word and additionally incorporated attention mechanism to capture contextual information.

Similar to Boudin, Nie, and Dawes [23], Jin and Szolovits [96] also remove the need for hand-crafted features by introducing a BiLSTM to encode word2vec vectors [149] corresponding to words in an input sentence $s = w_1, w_2, \ldots, w_N$ and generate hidden representations h_i as shown in (4). They then use an attention mechanism originally proposed by Bahdanau, Cho, and Bengio [14] to measure the relevance of each word to the whole sentence (5), in order to form a final sentence representation s in (6) which is later used by a CRF layer for classification.

$$h_i = \text{BiLSTM}(w_i) \tag{4}$$

$$\begin{aligned} u_i &= \tanh(W_s h_i + b_s) \\ \alpha_i &= \frac{\exp(u_i^\top u_s)}{\sum_i \exp(u_i^\top u_s)} \end{aligned} \tag{5}$$

$$s = \sum_i \alpha_i h_i \tag{6}$$

Jin and Szolovits [95] replace the BiLSTM Boudin, Nie, and Dawes [23] use with CLMs, BERT and BioBERT for encoding word vectors with small perturbations. Perturbing word vectors sent to a NN is an idea proposed by [202] to make NNs more robust to wrong or fake input and also improve their performance, a method commonly referred to as "Adversarial training" [68]. Besides the benefits of the contextualization using BERT and BioBERT, they reported improvements that adversarial training had in predicting P,I,O labels.

SLC to identify sentences in which outcomes are mentioned as achieved by all the above works is important. However, it is imperative that clinicians are able to effectively search and identify individual outcomes themselves especially considering that standardised terminology to describe them is gradually being introduced such as the taxonomy proposed by Dodd et al. [55]. This work therefore builds on work by Nye et al. [161] that recast the OD problem to additionally identify spans of text that correspond to outcomes through Token-Level Classification (TLC) as described in next section.

2.3.4 Token-level Classification (TLC)

While SLC achieves class or label prediction for a given sentence, TLC achieves class or label prediction for a single token [117, 207]. TLC spans across a huge range of NLP downstream tasks that are aimed at information extraction such as NER which identifies different types (such as names, location etc) of entities mentioned in text [67, 207] and Slot Filling (SF) which extracts certain attributes (or slots) of entities, which may be either persons or organizations [201].

TLC for OD is a sequence labelling task where LMs extract spans describing outcomes in RCTs. Nye et al. [161] performs two tasks including the extraction of spans that describe P, I or O elements and classification of each of these spans into a MeSH descriptor that they had respectively hierarchically classified into one of P, I and O labels. They use both a CRF and a BiLSTM-CRF to achieve automatic span tagging using train, dev and test splits of their EBM-NLP corpus that was described at the end of Section 2.3.2. The BiLSTM-CRF model outperformed the CRF in the PIO span extraction task despite the latter consuming more features including adjacent words to current word, POS features and character information such as upper or lower case information on tokens. However the CRF outperformed a logistic-regression model using n-grams in the span classification task.

BiLSTM-CRF model proved to be superior to other models such as CRFs and Logistic regression models in PIO span extraction and more specifically outcome span extraction or OSD in earlier TLC works.

The effectiveness of the LSTM-CRF models was further documented by Kang, Zou, and Weng [101] when they used it for PIO span tagging task on their gold standard set of 170 abstracts. Brockmeier et al. [26] slightly modify the BiLSTM-CRF by introducing an initial embedding layer which represents a token by concatenating a word vector

and a max-pooled vector over a set of character vectors processed by a BiLSTM. Additionally, they introduce BioBERT to encode the token embeddings. This modified model obtained the new SOTA performance in the PIO span extraction task using the EBM-NLP corpus before later being outperformed by Beltagy, Lo, and Cohan [18] and subsequently by Abaho et al. [2] particularly for outcome span extraction or OSD.

Similar to Brockmeier et al. [26], I take full advantage of pre-trained CLMs such as BioELMO [97] and BERT variants when tackling the OSD. The main motivation of this being that, the recent upsurge in performance in several NLP downstream tasks such as NER and RE has been attributed to the context-aware nature of these LMs enforced by their self-attention mechanism [54, 167].

2.4 NOISE REDUCTION IN BIONLP DATASETS

Curating qualitative datasets to facilitate the training of ML systems is a perpetual requirement considered by most researchers and industry data scientists. The difficulty with fulfilling this requirement is that, expert human annotators are not only scarce (especially for specialist and low-resource domains), but they are also expensive to hire.

For many NLP tasks, researchers are increasingly adopting crowdsourcing as a data collection strategy. Suhr et al. [198] scanned through proceedings of three top NLP conferences, ACL, EMNLP and NAACL and discovered that 6776 papers mention direct employment of crowdsourced workers. The advantage with crowdsourcing is, it is applicable to a diverse set of tasks ranging from QA [41], textual entailment [224], commonsense reasoning [184] and many more. The protocol in the crowdsourcing strategy is to 1) provide training (if necessary) and a set of heuristics (usually mandatory) that will serve as instructions to crowd workers and 2) provide a platform or tool on which the annotation or data collection task can be completed, such as Amazon Mechanical Turk[6], CrowdFlower[7] and BRAT[8].

For some tasks such as NER and RE, DS or weak supervision is a more popular approach to generating large amounts of labeled data. The goal is to automatically assign dataset samples labels based on some externally observed relatable facts or knowledge in an existing database, dictionaries, gazetteers or KBs which are often incomplete [178, 214]. For instance, in tasks such as RE, the intuition behind DS is that, if a KB specifies a relation existing between a pair of entities that have been identified in a sentence, then that is evidence of a relation between the entities and therefore, the sentence is labelled with the corresponding relation (in the KB) or as a positive mention [90]. Both resolutions, crowdsourcing and DS, are however unreliable and more often produce noisy annotations or otherwise weakly labelled data [90, 118, 204].

Crowdsourcing annotation using lay annotators and automatic annotation using Distant Supervision are unreliable and often produce noisy annotations or weakly labelled data.

6 https://www.mturk.com/
7 https://visit.figure-eight.com/People-Powered-Data-Enrichment_T
8 https://brat.nlplab.org/

To diminish the negative effects of this noise, recent DS approaches are incorporating noise filtering functions such as classifiers to remove noisy instance from the training data e.g. through a probability threshold [92] or a reinforcement agent [158]. Similar to how various works address the challenge of imbalanced classification labels, Le and Titov [119] re-weight dataset instances according to their probability of being correct and or noisy, and or an attention score [81, 119].

In the above scenarios, the noisy instances are detected and discarded. However, sometimes, rather than learn directly from noisy data, a noise model is introduced to gravitate the predicted noisy distribution towards the clean distribution during testing [141, 237]. Learning from partial annotations has been considered in order to detect false negatives, thus addressing the issue of incompleteness in DS datasets [233]. Chen et al. [35] adopts a group of reinforcement agents to relabel noisy instances.

Denoising weakly labelled data in BioNLP has garnered very limited attention, but nonetheless, a few works in the past have addressed this obstacle as narrated below.

To restore the intended structure of anonymised ophthalmology documents, Siklósi and Novák [191] use a semi-automatic approach that involves string matching to identify patterns of hand crafted features such as POS, date stamps, white spacing and character cases with the ideal patterns. A high cosine similarity is obtained between sentences corrected by this approach and those corrected by a human, with sentences being represented using Term frequency–Inverse document frequency (Tf-Idf), a statistical measure that evaluates how relevant a word is to a document in a collection of documents [6].

To address noise in a DS RE dataset, Li, Wu, and Vijay-Shanker [126] propose three heuristics, Closest Pairs (CP): which retained the closest pair of entities (with shortest path length in the dependency path) amongst multiple similar entity-pair mentions within a sentence, Trigger word (TW): which discarded positive instances whenever the stem of a trigger word was not found on the shortest dependency path and High confidence patterns (HP): which identifies and removes negatively labelled instances with trigger words on their shortest dependency path.

Similar to Li, Wu, and Vijay-Shanker [126], Intxaurrondo et al. [90] curb noisy relation labels using three heuristics that include (1) discarding high frequency positive and negative relation mentions ($>$ 90), (2) discarding relation mentions with a very low Pointwise Mutual Information (PMI) ($<$ 2.3) between its corresponding pair of entities, PMI between a pair of words $\mathrm{PMI}(x, y) = \log_2 \frac{P(x,y)}{P(x)P(y)}$ is a measure of the strength of association or co-occurrence of x and y against their independent occurrence or chance, and (3) retains rela-

tion mentions with a high cosine similarity ($\leqslant 90\%$) with the relation label centroid.

Takamatsu, Sato, and Nakagawa [204] incorporate a generative model in the DS labelling process to maximise $p(relation|pattern)$, the probability that a certain pattern expresses a particular relation. They model a probability $b_{rs} = P(x_{rsi}|Z_r, a_r, d_r, \lambda)$ with which DS assigns a relation r to an entity pair i appearing in pattern s given a binary variable z_r which is 1 if s expresses relation r and a_r, d_r and λ are all learnable parameters.

Xu et al. [232] use SVMs [171] to assign a probability of relevance of each negatively labelled relation and thereafter selects the top ranked relation mentions to add to the training set.

2.5 JOINT TOKEN- AND SENTENCE-LEVEL CLASSIFICATION IN BIONLP

Multi-task learning (MTL) has scaled the heights of NLP by proving that, several different tasks can simultaneously get done with accuracy and precision, something which would be enormously difficult for humans, if not impossible, to achieve [62, 180]. Using the same set of model parameters for a Multi-task learned system to generalize across different but related tasks can be rewarding, especially because the model would have benefited from learning the correlations across these tasks [46, 174]. Under the guise of MTL, joint learning of a dichotomy of tasks has been steadily progressing within the NLP community. Utilizing token-level and sentence-level information, Ma et al. [143] target joint slot filling, a NER TLC task and intent classification, a sentiment SLC task. Xu and Sarikaya [231] target joint intent recognition, a topic SLC task and entity classification, a NER TLC task. Xu et al. [230] and Karimi, Rossi, and Prati [103] achieve joint Aspect extraction, a NER TLC task and Aspect classification a sentiment SLC of customer reviews. Several authors attempt joint NER and RE using various datasets [17, 36, 87, 247]. Below is a discussion of previous efforts in joint learning strategies, particularly works pertaining to biomedical information extraction.

Using the same set of model parameters to generalise across multiple tasks can be very rewarding especially because, the MTL model would benefit from learning the correlations across multiple tasks.

A few NLP authors have exploited the mutual relationships (which can be probabilistically modelled) that exist between related but different biomedical information extraction sub-tasks such as Disease Named Entity Recognition (DNER) and Disease Named Entity Normalisation (DNEN): where the former aims to extract granular disease names and the latter aims to map disease entities to standardised vocabulary concepts, Medical Named Entity Recognition (MNER) and Medical Named Entity Normalisation (MNEN): where the former aims to extract medical named entities and the latter aims to map medical entities to a standardised vocabulary concepts.

Zhao et al. [247] overcomes the huge dependency on the hand crafted lexical and linguistic features used in transition-based models based on markov property [61] to jointly score MNER and MNEN [120, 137] by proposing a neural MTL framework stacked with an embedding layer, a CNN and a BiLSTM. Their neural MTL framework used a BiLSTM to encode character level CNN word embeddings concatenated with pre-trained word2vec word embeddings [149].

Earlier works were reliant on hand crafted lexical and linguistic features when jointly learning tasks such as DNER and DNEN.

Ji et al. [91] combined Zhao et al. [247]'s MTL framework with transition-based models such as [120, 137] to achieve joint DNER and DNEN. Transition-based models construct structured output by incrementally transitioning from one state to another such as in markov models [61], for instance, in a sequence labelling task like DNER that takes an input sequence of tokens, they formalize a tuple (S, A, c_s, C_t), where S is a set of states for all tokens, A is a set of possible actions to calculate probability of decoding the right output in a give state, c_s is an initial state or initial token position, and C_t is a set of terminal states. Ji et al. [91] initializes tokens as a concatenation of GloVe and ELMo embeddings. They use BiLSTMs and StackedLSTMs to generate sequence representations that capture transitioning from one state to another, in which case each token can assume either one of three states, stored (σ), processed (β) and output (O). They train a model to maximize the probability of predicting the correct action given a particular task based state representation (DNER (8) or DNEN(9)) as shown in (10). Five possible actions were catered for which involve transitioning across all three states.

$$\begin{aligned} b_t &= \text{BiLSTM}([\beta_0, \beta_1, \ldots]) \\ s_t &= \text{StackLSTM}([\ldots, \sigma_1, \sigma_0] \\ a_t &= \text{StackLSTM}([a_{t-1}, a_{t-1}, \ldots]) \end{aligned} \tag{7}$$

$$r_t^{NER} = \text{RELU}(W[s_t^1; s_t^0; b_t^0; a_t^{-1}] + d) \tag{8}$$

$$r_t^{NORM} = \text{RELU}(W[l'_m; r'_m; m'; c'; c; a_t^{-1}] \tag{9}$$

$$\underset{A,S}{\text{argmax}} \prod_t p(a_t | r_t) \tag{10}$$

Joint NER and RE is another joint learning task that has previously been attempted in BioNLP as well as other NLP specializations. Both Miwa and Sasaki [153] and Li et al. [125] encode entity features using a BiLSTM, however they differently encode relation features with the former using Tree-LSTMs [203] and the latter using a BiLSTM. A Tree-LSTM is a variation of an LSTM which generates a hidden state

from an input vector at a current time step and the hidden states of all its children nodes from a tree structure of the entire input sequence. Tai, Socher, and Manning [203] prove that the Tree-LSTM is able to propagate both the order sensitive sequential information an LSTM provides and structural information from a tree structural representation (such as a dependency tree) of a sentence. The limitation with this approach is the reliance on hand crafted features such as the dependency tree information. To alleviate this problem, Katiyar and Cardie [104] and Bekoulis et al. [17] propose stacked neural models that encode entire sequences, word-by-word including non-entity and non-relation spans.

Katiyar and Cardie [104] add an attention layer over a BiLSTM processing token embeddings, and use a softmax to decode both NER and RE labels. Instead of a softmax, Bekoulis et al. [17] uses a CRF for decoding NER labels and sigmoid function for multi-relation prediction, i.e. it computes multiple joint probabilities of a head entity and relation label pairs for each token. Essentially, the RE labels are not treated as mutually exclusive and therefore, multiple head-relation pairs can be predicted for a single token.

Chen et al. [36] replace the BiLSTM in [104] and [17] with pre-trained transformer models BERT and BioBERT. They use both the label distribution from the NER classification head as well as the encoded token representation from previous layer (BERT/BioBERT) in two different ways, (1) add up the two and use the resultant representation for predicting a single head-relation pair using a softmax layer and (2) use a biaffine attention mechanism [58] that allows the two vectors to interact and produce a vector used to predict a relation for each token. Without access to dependency trees and POS tags, their model shows an improvement over standalone models (achieving NER and RE independently) in the RE experiments.

Aside from BioNLP, some NLP works have demonstrated the effectiveness in joint learning strategies that combined separate but related tasks. Karimi, Rossi, and Prati [103] and Xu et al. [230] undertake a machine reading comprehension (MRC) task called Aspect Based Sentiment Analysis (ABSA) which extracts aspects from customer reviews and classifies them into corresponding opinions or sentiments. They perform ABSA by feeding BERT with a sentence $s = ([CLS], x_{1:j}, [SEP], x_{j+1:n}, [SEP])$, where $x_{1:j}$ is a sentence containing an aspect of a product, $x_{j+1:n}$ is a customer review sentence directed to the aspect and [CLS] is a token not only indicating the beginning of a sequence, but also a sentiment polarity in the customer review about the aspect. They fine-tune a BERT model to conduct both aspect extraction and aspect sentiment classification.

Besides joint learning tasks in BioNLP, joint learning in NLP has been used for tasks such as ABSA, i.e. joint aspect extraction and aspect classification.

The above mentioned works tend to generate attention-based sentence-level representations that encapsulate the contribution each word would make in predicting sentence categories. I however propose a joint

learning strategy that generates attention-based representations at both word- and sentence-level to be respectively used in predicting word- and sentence-level categories/labels.

2.6 PROMPT BASED LEARNING FOR TEXT GENERATION IN BIONLP

Prompt based learning (PBL) emanates from the idea of reformulating downstream tasks to look more like those solved during original LM training with the help of a textual prompt [135]. While traditional supervised learning models the probability $P(y|x;\theta)$, where y is a target label provided in a labelled dataset, PBL models the probability $P(x;\theta)$ of text x itself and thereafter uses this probability to predict y which is often text. PBL's set up in which LMs encode and answer question-like formatted sentences such as "Eifel tower is located in the town of ___", has precipitated the notion that suggests a LM can be treated as an alternative to, or at least a proxy for a KB. This said, by virtue of the question-like formats, prompting can inherently achieve information extraction such as, given a prompt like "No reason to watch, it was ___ movie", recent works train a LM to predict a word such as horrible (a negative sentiment) to suitably fill in the blank [66, 140, 151].

Probing factual knowledge in PLMs using prompts has been a success in the general-domain i.e. using datasets that are not necessarily representative of any particular domain.

On multiple occasions now, probing factual knowledge in PLMs using prompts has been a success in the general-domain i.e. using datasets that are not necessarily representative of any particular domain [93, 94, 170, 186]. Five prompt training strategies commonly adopted include (1) Promptless fine-tuning, in which the PLM is not fine-tuned with prompts but rather gets parameters updated via gradients induced from downstream training examples [80], (2) Tuning-free prompting, in which the PLM is used off-the-shelf to generate answers to a prompt without updating its parameters [27, 170], (3) Fixed-LM Prompt Tuning, where the introduced prompt parameters are the only ones that get updated and those of the PLM do not [128], (4) Fixed-prompt LM Tuning, where the PLM parameters get updated but the prompt-relevant parameters do not [186, 187] and finally (5) Prompt+LM Tuning, in which both the PLM and the prompt-relevant parameters are updated all together [49, 93].

To this end, there has been little attention to leveraging the power of domain-specific PLMs to act as domain-specific KBs that can be queried for facts. However, Sung et al. [200] recently released BioLAMA, a benchmark dataset comprising 49K biomedical factual knowledge triples (curated from digital archives including CTD[9], UMLS[10] and Wikidata[11]) that can be used for probing biomedical PLMs. BioLAMA contains relational triples reconstructed as fill-in-the-blank

9 http://ctdbase.org/
10 https://www.nlm.nih.gov/research/umls/
11 https://wikidata.org

cloze statements (or prompts). Despite the prompt reconstruction, results of fine-tuning using BERT [54], BioBERT [121] and BioLM [124] generally suggested the need for stronger biomedical LMs and probing methods.

Similar to traditional supervised learning, PBL suffers the limited training data bottleneck. To address this issue, methods that perform prompting in the embedding space of a model have emerged [128, 173, 190]. Shin et al. [190] use a gradient based strategy to automatically create continuous/soft prompts to cause a MLM to produce desired knowledge. A set of 5 trigger tokens "[T][T][T][T][T]" are each initialised as a [MASK] token and added to sentences used in downstream tasks such as Sentiment Analysis, then iteratively these trigger tokens are swapped with a vocabulary token while trying to maximise the label likelihood in the downstream task. As shown in Equation 11, the top k tokens (for each trigger position) estimated to cause greatest increase are used to replace the trigger tokens,

$$\mathcal{V}_{cand} = \underset{w \in \mathcal{V}}{\text{top}-k}[w_{in}^{T} \nabla \log p(y|x_{prompt})] \tag{11}$$

Qin and Eisner [173] slightly changes the approach proposed by Shin et al. [190], by replacing trigger tokens with arbitrary vectors $\{v_i\}_{i=1}^{i=5}$, and introducing a small perturbation vector $\triangle$ that is initialised to o and then iteratively added to the arbitrary vectors when fine-tuning the model and prompt parameters. A mixture modelling framework is used to generate the soft prompts as shown in Equation 12 later used in the downstream evaluation.

$$p(y|x,r) = \sum_{t \in \mathcal{T}_r} p(t|r) \cdot \text{PLM}(y|t,x) \tag{12}$$

Upon evaluation of the soft prompts in downstream tasks, Qin and Eisner [173] soft prompt model consistently improves performance of Shin et al. [190] on the relations dataset T-REX [63], LAMA on Google RE dataset[12].

Several works on PBL still heavily rely on handcrafting a linguistic pattern or shape that prompts should take on.

In-spite of all the efforts to outgrow the reliance on training data for PBL, several works including the gradient based approaches above [173, 190] still heavily rely on handcrafting a linguistic pattern or shape that prompts should take on. This constraint is enforced in order to place a lot of emphasis on subject-relation-object triples when fine-tuning LMs on prompts. The challenge with the constraint is, the search space of possible linguistic patterns prompts can take on is enormous and it is therefore practically infeasible to reformulate prompts into all possible patterns. My efforts are motivated by the fact that, there should be minimal or no need at all to worry about

12 https://github.com/google-research-datasets/relation-extraction-corpus.

the pattern of the prompt, but rather, we can leverage information local to the prompt such as word positions. I attempt to enhance a words contextualised representation with position based representations to capture the words position relative to the mask to be filled. Previously some works have used similar position-aware attention over LSTMs for relation extraction, sequence labelling and slot filling tasks in different datasets [220, 245]. To the best of my knowledge, I pioneered the use of an extra position-attention layer above transformer models such as BERT to solve the fill-in-the-blank prompting task.

It is important to note that, the PBL task I conduct is more closely aligned to prompting for information extraction purposes. Nonetheless, I explore a few-shot learning setup in which I only assume a few annotated examples and therefore explore the capacity of the PLM to generate outcomes.

2.7 BIONLP

Similar to the widely acknowledged collection of benchmark tasks and datasets facilitating language understanding tasks in general-domain NLP research known as "General Language Understanding Evaluation" (GLUE) benchmark [215], there has been a number of recurring tasks demanding attention in BioNLP research. Recently, Gu et al. [70] established BLURB (Biomedical Language Understanding & Reasoning Benchmark), a comprehensive set of BioNLP tasks from publicly available datasets to help accelerate progress in clinical NLP research. This section provides descriptions of the most commonly tackled BioNLP tasks, their benchmark datasets and the progress they have made.

2.7.1 Named Entity Recognition (NER)

2.7.1.1 Chemical (drug) and Disease detection

Through BioCreative V (a community challenge event for Biomedical text mining) [83], Wei et al. [219] designed two tasks, DNER, to extract diseases and Chemical-Induced disease relation extraction (CID) to extract chemicals-disease relations from 1500 PubMed articles manually annotated for diseases and chemicals (drugs). Li et al. [127] propose BioCreative V Chemical-Disease Relation corpus (BC5CDR) to support DNER and CID tasks. BC5CDR consists of 1500 PubMed articles (with train, development and test splits) expertly annotated for chemicals, diseases and chemical-disease interactions. Courtesy of the recently released CLMs premised on the transformer architecture [213], SOTA performances have been achieved on this dataset and sim-

ilar datasets (NCBI disease corpus [56]) for DNER, CID and chemical recognition [18, 54, 70, 242].

2.7.1.2 Gene and Protein Mention

Automatic identification of strings that correspond to gene and or protein name mentions in clinical articles dates back to the early 2000s [205, 238]. The BioCreative II Gene Mention corpus [193] is the popularly used benchmark dataset for the Gene Mention task. The corpus was annotated using expert guidelines, and originally contained 20000 sentences split into 15000 and 5000 for training and testing respectively. However, newer versions that portioned off 2500 sentences for development from the training set have been introduced [48]. The Genia corpus [162] comprising 2000 Medline[13] abstracts [155], on the other hand is commonly used in evaluation of tasks detecting mentions of molecular biology entities including protein, DNA, RNA, cell line and cell types [106]. Similar to the performance trend on DNER and CID tasks above in Section 2.7.1.1, performance on both the gene and protein mention tasks has been dramatically improved by both CLMs and knowledge enhanced LMs [70, 242].

2.7.1.3 De-Identification tasks (de-ID)

Focused at preserving patients confidentiality and privacy whilst sharing health information, de-ID task automatically finds and removes Personal Identifying Information (PII) from clinical records [57, 157]. PII categories may include but not limited to; Names (of patients or doctors or health-facilities), IDs (alphanumeric codes uniquely identifying patients or doctors or health-facilities), Dates, Locations, Phone Numbers and Ages. Earlier de-ID work used rule-based approaches like regular expression pattern matching to locate PII information referenced from look-up dictionaries, and then replace it with tags indicating the corresponding PII category [71, 157]. Uzuner, Luo, and Szolovits [211] built a corpus for de-identification in which they subtly replaced PII with realistic surrogates (randomly selected meaningful character combinations). They then compared several rule-based and ML systems ability to differentiate PII from non-PII information at token- and instance- level. Unsurprisingly, because of their unique ability to detect complex patterns, ML systems significantly outperformed rule based systems in both these tasks. More so, hybrid strategies that employed regular expression features in the ML setups performed even better. Of late, CLMs pre-trained on clinical notes have proven to be superior to all prior methods in the de-ID task [10].

In terms of task formulation, the above mentioned BioNLP NER (sequence labelling) problems are most closely related to the tasks undertaken in this work, in particular, the OSD task. While the above

13 https://www.nlm.nih.gov/medline/medline_overview.html

tasks extract text spans that describe drugs, diseases, genes or proteins and PII, OSD extracts spans that describe health outcomes mentioned in RCTs. The datasets built to support the above tasks provide labels at the word level for each entity. Word-level labels are used in supervised learning for detection of text spans describing the aforementioned biomedical entities. In terms of differences, OSD has attracted less attention than the discussed NER tasks mainly because of the scarcity of publicly available annotated corpora to facilitate OSD [161]. This serves as a main motivation for the annotation of a new dataset as well as denoising outcome annotations in a publicly available dataset as discussed in Chapter 3. This dataset is subsequently used to train models to infer outcome span detections.

2.7.2 Relation extraction (RE)

2.7.2.1 Drug-Drug Interactions (DDIs)

Drug-Drug Interaction (DDI) occurs when one drug influences the activity or level of another drug [188], and for the matter, DDIs in BioNLP refers to the detection and classification of interactions or otherwise relations between drugs in biomedical text. The first published dataset to support the DDI task is the DrugDDI corpus [188] containing 579 documents with approximately 10 sentences per document. A pharmacist was used to manually annotate DDIs within the documents in which drugs had been automatically recognised using the MetaMap Transfer tool (MMTx).[14] Because DrugDDI was annotated by single annotator and the automatically recognised drugs were never validated by an expert, Herrero-Zazo et al. [76] undertook several steps to improve its standards and introduce the DDI corpus. Improvements in DDI included, a further 446 documents from DrugBank [227] and Medline [155], pharmacodynamic (PD) and pharmacokinetc (PK) DDIs, a review of the automatically annotated drugs by two expert pharmacist annotators, a list of annotation guidelines and finally an inter-annotator agreement to validate the consistency and quality of annotation.

2.7.2.2 Chemical-Protein Interactions (CPIs)

CPI task aims to detect and classify interactions or relations between chemical and protein (or gene entities) from biomedical literature. Similar to DDI above, CPI plays an important role of understanding molecular mechanism of adverse drug reactions. Built during the BioCreative VI challenge [146], ChemProt corpus has been used to facilitate biomedical text mining for CPIs. It is exhaustively manually annotated for chemicals (drugs), proteins and their relations which

14 https://lhncbc.nlm.nih.gov/ii/tools/MetaMap.html

are of 22 different kinds. Lim and Kang [129] achieved relative success in detecting CPIs using a tree-LSTM model [203] and PubMed-and-PMC-word2vec embeddings[15] augmented with positional and sub-tree containment features. The positional feature represented the relative distance from each word to the target entities, whereas the sub-tree feature was calculated in the parsing state and it indicated that a certain sub-tree contained a target entity. Several SOTA performances have been achieved when researchers have addressed the problem with CLMs pre-trained on biomedical text [70, 242].

2.7.2.3 Gene-disease Interactions

Because of the role genetics plays in the development of diseases, it is imperative that clinical researchers understand the links between genes and human diseases [163]. For that matter, automating ways of extracting associations between genes and diseases from biomedical literature is crucial. Early efforts in resolving the problem, involved counting the co-occurrence frequencies of genes and diseases in articles, and if the frequency of a co-occurrence of a particular gene-disease association was significantly higher than a certain expected threshold, then that association was considered valid [5]. The surging interest of the BioNLP community in this task has seen an introduction of new datasets and methods to address the problem. Several authors have used SVM to classify sentences into positive (contain gene-disease association) or negative (do not). For example, Özgür et al. [163] used both a dependency parser (to cut out paths between gene entities in sentences) and an SVM to capture 95% of the top 20 genes related to prostate cancer. Van Mulligen et al. [212] used five expert annotators and a NER based system to annotate drugs, diseases, genes, gene variants and the relationships between these entities, in 300 carefully selected PubMed and Medline abstracts [155], there after provided inter-annotator agreement details to support a corpus titled EU-ADR. Bravo et al. [25] build an even larger corpus with gene-disease associations from the Genetics Association Database (GAD) [156] and used kernel-based approaches to extract multiple genes linked to depression.

BioNLP relation extractions tasks, and in particular the ones discussed above were traditionally performed by directly classifying each candidate instance (e.g. a pair of drugs) into one or more predefined classifications. For instance, Björne, Kaewphan, and Salakoski [20] uses a SVM to classify drug-to-srug interactions. Progressively, DDI has been decoupled into two separate sub-tasks i.e. recognition of DDIs and classification of DDIs. Some authors have performed the two in tandem i.e. identify DDIs and subsequently classify the interactions [43, 108]. Similarly, I cast the OD task as two sub-tasks in

15 https://bio.nlplab.org/

Chapter 5 i.e. OSD and OC which respectively detects outcomes spans and then classifies the spans. Motivated by joint learning approaches such as joint extraction and classification of aspects in customer reviews [103], I proceed to build a neural model that jointly performs the two sub-tasks as later discussed in Chapter 5.

2.7.3 Reading and Comprehension

2.7.3.1 Biomedical Question Answering (BQA)

BQA stems from the need to synthesize and filter information from multiple sources of biomedical information. For instance, in order to obtain an answer to a question input into a digital clinical archive or search engine, clinicians have to (1) narrow down a list of results to retain relevant articles or structured text and then (2) combine the relevant text, study it and filter out answers they may seek [208]. BQA is therefore aimed at directly producing answers to biomedical questions posed by clinicians, for instance, *"What is the most common condition related to sleeplessness and fatigue?"*, whilst a BQA system may not return the golden answer *"Insomnia"*, it can return an ideal answer such as *"Typically waking too early, failing to fall back asleep and spending a lot of the night lying awake are commonly experienced by Insomnia patients"*. The BioASQ [208] challenge prepared a biomedical Semantic QA task in which participants built systems to annotate questions with concepts from relevant ontologies and depending on the type of the question, the ideal answer would either be exact or paragraph-sized. [98] is the other commonly used benchmark dataset for the BQA task. It contains questions annotated with "yes, maybe and no" answers. Results of the evaluation on these two datasets show that human answer baselines outperformed a Support Vector Regression model (on the BioASQ dataset) and fine-tuned BioBERT (on the PubMedQA dataset).

2.7.3.2 Medical Natural Language Inference (MedNLI)

MedNLI is a variant of the NLI task specific to the clinical domain. NLI is aimed at determining whether a given hypothesis can be inferred from a given premise. Romanov and Shivade [179] employed four clinicians to annotate a total of 14049 sentences with anonymous patient records extracted from MIMIC-III v1.3 database [99]. Using prompt instructions, clinicians wrote three different alternate sentences for each original sentence in the dataset, where one was definitely a true description of the original, another was a probably true depiction of the original and the third was definitely a false description of the original. After mapping each sentence with at least one of the 3 alternatives to obtain sentence pairs, the goal was to build a model that would classify each pair of sentences into 3 dif-

ferent classes i.e. entailment, contradiction and neutral. InferSent [47] that uses a pair of BiLSTM encoders (each word represented by a fastText embedding [21]) and a self attention layer to generate universal sentence representations outperformed a Bag-of-words (BOW) model and ESIM [37], a model with a chain of LSTMs, that encodes and max-pools over the LSTMs output before classification. Later on, the CLM ClinicalBERT [10] has obtained SOTA results on the MedNLI dataset.

Despite being distantly related to OD, the above tasks and in particular BQA have been achieved using sequence-to-sequence models [223, 239] in which an input question is encoded by one model (encoder such as BiLSTM) and the output produced by another model (decoder such as BiLSTM). Similarly, I adopt a sequence-to-sequence architecture for the joint learning method proposed to simultaneously achieve OSD and OC in Chapter 5.

2.8 DISCUSSION AND SUMMARY

The comprehensive review of the prior research efforts representative of EBM NLP, and in particular OD has revealed both the significant progress achieved to this day as well as the ton of work that still has to be done. In summary, the chapter aimed to survey the following: *how prior work defined OD and what resources (datasets) currently exist to support OD, prevailing challenges encountered when conducting OD, conventional and modern methods that have been used to achieve OD and finally, common BioNLP tasks related to OD*.

To answer these questions, the chapter systematically reviewed OD as a SLC as well as a TLC task. For the latter, prior research developed models to extract text spans that describe outcomes from larger bodies of text such as RCTs, and for the former, the goal was to develop models that would classify an abstract sentence as an outcome statement or not. Generally, earlier works heavily relied on using hand-crafted features and conventional ML methods such as NB and SVM implemented in ML toolkits, however in the recent past, research works including this thesis have advanced to neural LMs that model the probability of individual words in order to tackle the OD problem. In addition to an elaborated discussion on the use of pre-trained LMs in TL setups for EBM NLP tasks, the chapter reviews MTL approaches to jointly achieve tasks such as NER and RE. Furthermore, the chapter reviews prompting methods used to probe for KB facts or information (such as in information extraction) from LMs. The chapter concludes with a summary of BioNLP tasks related to OD.

The chapter has essentially motivated all the technical chapters of this thesis. To address the gaps and challenges in prior efforts such as flawed or noisy outcome annotations, the next chapter proposes a couple of denoising techniques to improve the evaluation performance in the OD task.

3

REFINEMENT AND ANNOTATION OF OUTCOME DATA

3.1 INTRODUCTION

The performance of NLP systems is massively influenced by the quality of annotations in training datasets [222]. To put it another way, weakly annotated data is more capable of degrading the performance of NLP systems than strongly or expertly annotated data [16, 126, 222]. Performance degradation attributable to weak annotations is not just empirically encountered, but it further misinforms and misleads the consumers of these systems. Highly specialised domains such as BioNLP are prone to this problem because, dataset curators (annotators) with biomedical knowledge are few and expensive to hire. To avoid the expense, researchers resort to crowdsourcing annotations which involves employing lay curators to annotate data. On other occasions, DS offers a viable approach to produce labeled data [152, 178, 214]. DS involves automatically assigning dataset samples labels based on some externally observed relatable facts or knowledge in an existing KB [152, 178, 214]. Both resolutions are however unreliable and will often lead to noisy annotations. Lay annotators are not quite competent to deal with the complex nature and domain specific terminologies in biomedical literature [1], whereas, with DS, the KBs used do not cover all existing knowledge about a subject i.e. they are often incomplete [126, 204] and additionally, rule-induced systems (such as DS) can be erroneous [12].

To address noise in annotations, I propose a framework that automatically corrects flawed outcome annotations, an unsupervised label denoiser and introduce a new expertly labelled dataset of outcomes.

To address the challenge of noisy annotations described in the previous paragraph and reviewed in Section 2.4 of Chapter 2, this chapter (1) Proposes a framework that automatically corrects incorrectly captured annotations of outcomes, using EBM-NLP corpus [161] as a case study in Section 3.2, (2) Introduces a novel outcome dataset, EBM-COMET, in which outcomes within RCTs are expertly annotated with outcome classifications drawn from a standardised outcome classification system [55] in Section 3.3 and finally, (3) Introduces a label denoising approach that uses unsupervised text alignment of labels in comparable datasets in Section 3.4. This alignment approach is later used for data augmentation in a low-resource setting in the work covered in Chapter 5. It is important to note that, whilst the EBM-NLP corpus has Participants (P), Interventions (I) and Outcomes (O) annotations, the propositions made in this chapter are focused on the Outcome (O) element.

3.2 DENOISING CROWDSOURCED ANNOTATIONS OF OUTCOMES

This chapter investigates a recently published corpus, EBM-NLP [161], comprising ca. 5000 abstracts annotated with P, I, O elements by a mixture of lay and expert annotators. The corpus has two versions, (1) the "**starting spans**" in which text spans are annotated with the literal "P, I or O" labels and (2) the "**hierarchical labels**" in which the annotated "P, I or O" spans were further annotated with more specific labels aligned to the concepts codified by the MeSH[1], for instance the "O" spans are annotated with more granular (specific) outcome type labels which include Physical, Pain, Mental, Mortality, Adverse effects and Other. An outcome type is a classification or category that collectively embodies a group of outcomes measured during clinical Trials [55].

Focused on the outcome element, this investigation begins with an assessment of whether the annotations retain the true identity of a widely acknowledged definition of an outcome i.e. a measurement or an observation used to capture and assess the effect of a treatment such as assessment of side effects (risk) or effectiveness (benefits) [55, 225]. For this assessment, I rely on two domain experts in order to eliminate traps such as annotation bias that prior construction efforts encountered [84, 85, 214]. Annotation bias occurs when annotations of the same data vary from one annotator to another as a result of ineliminable factors such as background, preconceptions about the data and knowledge level of the annotation task [11]. After obtaining a very low inter annotator agreement in an annotation exercise, Boudin, Nie, and Dawes [23] resorted to weakly labelling sentences with the explicit headings under which they were mentioned in the PubMed articles.

In this assessment exercise, the experts review a small sample of annotated abstracts checking whether the annotated outcomes retain the identity of an outcome as defined above, and if not, they determine the flaws that recur across the annotations in the sample. By the end of this review, multiple flaws within the annotations had been carefully identified, and these are outlined and discussed in the following Section 3.2.1.

A flaw is an error or mistake in an annotation usually resulting from human faults during manual annotation of data.

3.2.1 Flaws discovered in annotations of health outcomes

A flaw can be defined as an error or mistake in an annotation, usually resulting from human faults during manual annotation of data.

Below is a breakdown of the different flaws observed in outcome annotations in the review conducted as previous paragraph discusses.

1 https://www.nlm.nih.gov/mesh/meshhome.html

Flaw 1: Inclusion of unnecessary text that is either supportive of the actual outcome or an elaborated context of an outcome. Two kinds of unnecessary text identified and presented in Table 3 are,

1. Statistical metrics.
   Statistical terms such as *mean*, *median*, *standard deviation* are relevant in reporting results but are not considered as outcomes themselves.
2. Modifying or descriptive Part-Of-Speech (POS).
   Comparative POS such as adjectives, conjunctions and adverbs were captured as part of the sequence of words in outcome spans. E.g. "*Lower*" in the phrase "*Lower maternal attachment*" can also be "*higher*" which are both comparative adjectives describing the change as applied to an outcome "*maternal attachment*".

Incorrectly captured Outcome	Correct Outcome
1. mean arterial blood pressure	arterial blood pressure
2. median Survival	Survival
1. Improved ADHD symptoms	ADHD symptoms
2. Lower maternal attachment	maternal attachment

Table 3: Examples of unnecessary text such as statistical and POS tags.

Incorrectly captured Outcome	Correct Outcome
1. cardiovascular events-(myocardial infarction, stroke and-cardiovascular death)	1. myocardial infarction 2. stroke 3. cardiovascular death
2. Systolic and Diastolic blood-pressure	1. Systolic blood pressure 2. Diastolic blood pressure

Table 4: Examples of multiple distinct outcomes compressed into one outcome.

Contiguous outcome spans incorrectly captured as a single outcome e.g. systolic and diastolic BP should be systolic BP and diastolic BP.

Flaw 2: Failure to identify independent or granular outcomes. This was observed across the following,

1. Multiple outcomes annotated as a single outcome.
   Some outcome spans were captured as a sequence of distinct outcomes syntactically separated by either logical con-

junctions (and/or) or punctuation characters such as commas, full and semi-colons (example 1 in Table 4).

2. Contiguous outcome spans annotated as a single outcome. These included outcome spans that depicted two or more distinct but related outcomes. e.g. *Systolic and Diastolic blood pressure* represents two different but related outcomes as shown in example 2 in Table 4.

Flaw 3: Capturing measurement tools, metrics and results as outcomes.
The phrase "*Work-related stress scores*" is a metric result reported during RCTs, but the outcome itself is "*Work-related stress*". Other examples may include tools such as questionnaires and tests used in RCTs. Examples are shown in Table 5.

Incorrectly captured Outcome	Correct Outcome
1. Quality of life Questionnaire	Quality of life
2. Work-related stress scores	Work-related stress
3. Weight-test	Weight

Table 5: Examples of measurement tools and metrics captured as outcomes.

Quality of life questionnaire is not an outcome but rather a tool measuring the outcome Quality of life. Similarly, Work-related stress scores is not an outcome but rather a score associated the outcome Work-related stress.

Flaw 4: Imprecise outcome annotations resulting from inadequate domain knowledge of annotators. Examples indicated in Table 6.

1. Non-outcomes incorrectly captured as outcomes e.g. "*Severity*", "*Effect sizes*", "*significant improvement*".
2. Misrepresented outcome types, especially in the Mortality outcome type.

outcome span	Incorrect Type	Correct Type
Nauseas and Vomiting	Mortality	Physical
suicidal ideations	Mortality	Mental

Table 6: Examples of outcomes labeled with incorrect types.

Flaw 5: Combining annotations of outcomes in non-human studies together with those in human studies. Despite the validity of outcomes in non-human species, they ought to be separately annotated. For example, *time needed to treat commercial beef cattle* is an outcome extracted from non-human medical abstracts included in outcome annotations for human medical abstracts.

3.2.2 A hybrid approach to correcting outcome annotations

I propose a novel noise filtering framework that combines POS heuristics and rule-based chunking to automatically correct flawed granular outcome annotations. The framework incorporates constraints that examine the syntactic and semantic structure of the annotations in order to reduce the noise identified as flaws in the above section. POS heuristics are concerned with assigning each word in the abstracts a POS tag which can be used to identify irrelevant words and characters that are captured as part of outcomes. Rule-based chunking is concerned with identifying and removing unwanted text from the outcome span using POS trigger tags accompanying actual relevant text (in outcome span). The components of this framework are described in the following sections and later summarized in Algorithm 1.

3.2.2.1 Custom Part-of-speech tagging

The proposed noise filtering framework uses a trained tagger customised using spaCY to annotate sentences and a chunker relies on the POS tags to identify flaws.

BioNLP is supported by a number of POS taggers such as MedPost/SKR Tagger [192] and Genia Tagger [209] which were all trained on biomedical text in Medline sentences [192]. Whereas these taggers would suitably perform POS tagging for the OD tasks, I opt to use a more universally recognised SOTA NLP industry-scale library, spaCY[2] for POS tagging. spaCY already has models pre-trained on web text (blogs, news, comments) to perform a variety of text pre-processing tasks including Tokenization, POS tagging, Dependency parsing, sentence segmentation and more. Additionally, spaCY provisions for customizing or updating[3] these trained components by fine-tuning them on domain-specific data.

To leverage the entire suite of inbuilt text-prepossessing components of this library, particularly Tokenization and sentence segmentation, I train a spaCY POS tagger on Medpost, the same corpus MedPOST tagger was trained containing 6,700 Medline sentences annotated with 60 POS tags [192]. This not only allows spaCY tagger to adapt and ably generalise well across biomedical text for the OD tasks, but it is additionally a less computationally expensive approach in comparison to loading two different models i.e. spaCY for tasks like tokenization and MedPOST for POS tagging.

The trained tagger is then subsequently used to assign POS tags to every individual word in the investigated dataset (EBM-NLP). The trained tagger conforms to Penn Treebank POS tagging guidelines [183], with a few adjustments that include,

- All words that ended with '+' such as CIN2+ were assigned noun tags, 'NN'. This catered for some medical compounds and

2 https://spacy.io/
3 https://spacy.io/usage/training

An illustration of the entire pipeline of the noise filtering and evaluation framework

substances with similar syntax that could have not appeared in the training set.

- Punctuation symbols such as period (.), single quotation (') and semi-colon (;) were eliminated because the EBM-NLP dataset had several of these as redundant punctuation tokens.
- Square brackets retained their syntax as the corresponding POS tag i.e. '[' and ']' were tagged as '[' and ']' respectively.

3.2.2.2 Rule-based chunking

The chunking algorithm (chunker) relies on a set of heuristics to determine where the chunk of interest (correct outcome span) begins and ends. These heuristics are handcrafted linguistic constraints created to influence the capturing of sequences of words relevant to an outcome within the incorrect crowdsourced outcome spans. Exposed to the POS tagged outcome spans from the previous step, this chunker uses underlying syntactical patterns known as regular expressions to programmatically extract one or more sub text-spans that constitute the actual outcome span of interest. For example, given a POS tagged outcome span such as *"lower_JJR maternal_JJ attachment_NN"* produced from Section 3.2.2.1, based on one of the predefined heuristics below that suggests removal of comparative POS such as comparative adjectives tagged 'JJR', the chunker uses the positional information of word tagged with the unwanted POS i.e. *"lower_JJR"* to strip it off and retain *"maternal attachment"* as the outcome. Below is a list of chunking heuristics (H) used,

H1. **Removal of statistical terms:** Statistical terms within outcomes were eliminated irrespective of their position in the outcome spans. The removed statistical terms were referenced from a couple of sources including the international institute of statistics glossary[4] and the book for medical device clinical trials [4].

H2. **Removal of non-informative POS tags:** These included all stop words with POS tags; **TO** (infinitive marker), **IN** (Preposition), **CC** (coordinating conjunction) and **DD** (determiner). Despite frequently occurring in text, stop words are considered to be non-informative and therefore deemed irrelevant in analysis [160]. Stop words were therefore removed whenever they were located at,

- Start or end of outcome spans. e.g ~~**the**~~**_DD** memory_NN loss_NN, ~~**and**~~**_CC** fatigue_NN.
- Every position in an outcome span, i.e. all words tagged with a mixture of only the above defined non-informative POS tags.

A set of heuristics that guide the noise filtration process.

H3. **Eliminating contextually comparative or quantification phrases:** Comparative phrases contain comparative adjectives and adverbs with POS tags **JJR** and **RRR** respectively such as *longer* and *better* respectively. Other phrases contain superlative adjectives and adverbs with POS tags **JJT** and **RRT** respectively such as *highest* and *most*. I additionally considered a set of terms depicting quantity and their synonyms extracted from WordNet [150]. These included total, average, increase and decrease. These were removed whenever they were at the start or end positions of an outcome span.

H3. **Removing unnecessary word sequences at the start of outcome spans:** Unwanted starting POS sequence included (**NNS II**), (**NNS DD**) and (**NNS TO**) e.g. *predictors of* is unnecessary in *predictors_NNS of_II sex_NN risk_NN behavior_NN*, and so is *changes in* in *changes_NNS in_II Brain_NN Natriuretic_NN Peptide_NN*.

H5. **Splitting long outcomes via logical conjunctions and punctuation characters:** These include coordinating conjunctions with a POS tag **CC** and punctuation such as commas with a POS tag ',' e.g. *Serum_NN folate_NN and_CC vitamin_NN B12_NN* is split at *and_CC* to have two seperate outcomes *Serum_NN folate_NN* and *vitamin_NN B12_NN*.

H6. **Stripping off square, curved or curly brackets wrapped around outcome spans:** For example the parenthesis in the span *(_(Autism_NN Behavior_NN)_)* is removed. The retained span would then be subjected to the heuristics outlined above.

H7. **Removal of measurement tools and metrics:** Referencing tools from the book for medical device clinical trials [4], I match measurement tools and metrics that were annotated as outcome spans and eliminate the last word in each span. For example, the last word of each of these tools i.e. Autism Behavior ~~checklist~~, Quality of life ~~questionnaire~~ and Baseline tumour ~~marker~~ are respectively removed.

H8. **Changing of an annotation label:** The outcome type (class labels) for the outcome spans that had incorrect labels were changed to the suitable class labels that the experts had provided during their assessment.

H9. **Outcome spans with a sequence of words all tagged as nouns were preserved:** For examples *platelet_NN thromboxane_NN formation_NN.* is preserved.

An algorithm used for the noise filtration process.

4 http://isi.cbs.nl/glossary/bloken00.htm

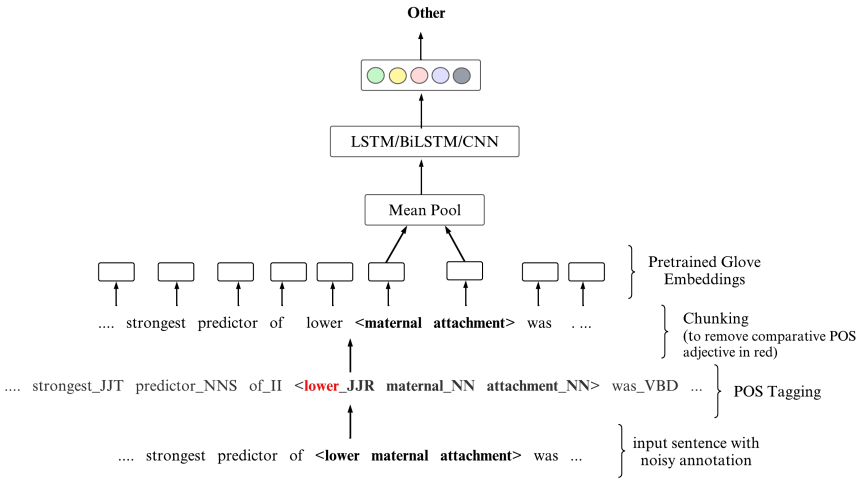


Figure 3: Outputs of the two main components (POS tagging and Rule based chunking) of the hybrid noise filtering framework and the architectures used in the outcome classification task in Section 3.2.3.

Algorithm 1 Noise filtering Algorithm

Input: Noisy outcome annotations NO, Heuristics {H1...H9} in Section 3.2.2.2,
Output: Corrected outcome annotations CO
for annotation x in NO **do**
  where $x = w_1, \ldots, w_n$
  for each w in x **do**
    ASSIGN POS tag to w
  end for
  ASSERT that x is POS tagged i.e. $x^{(p)}$
  for heuristic h in {H1...H9} **do**
    APPLY h to $x^{(p)}$
    UPDATE $x^{(p)}$ such as re-assign NONE (O) label to identified non-outcome words denoting it is irrelevant to actual outcome
  end for
  ASSERT that all $h \in H_{i=1}^{9}$ are applied to $x^{(p)}$ in order to obtain $x^{(c)}$
  Feed $x^{(c)}$ into CO
end for
return CO

3.2.3 Evaluation

To evaluate the methods proposed in the preceding section, I undertake an outcome classification (OC) task whose goal is to classify each outcome span into one of six outcome types (classifications) namely

Adverse-effects, **Mental**, **Mortality**, **Pain**, **Physical** and **Other**. These outcome types were originally prescribed as the outcome classifications in the hierarchical labels version of the EBM-NLP dataset introduced in Section 3.2.

TASK DEFINITION: Given a training-set, $O = \{(x_t, y_t)\}_{t=1}^{T}$, where T is the number of training examples, x_t is an instance of an outcome span such that $x_t = (w_1, w_2, \ldots, w_N)$ where N is the length of the outcome span sequence to be classified and each w_n is a 50-dimensional embedding for a word at index n. Word embeddings are obtained from pre-trained 840B 300d GloVe word vectors [166]. y_t is a one-hot vector corresponding to a label in the label space $\mathcal{Y}$. The goal is to learn a classifier that models the probability of each label for the incoming outcome span. The parameters of the classifier are trained to minimise the cross-entropy loss between the true and predicted distributions in (13),

$$L(y, p) = -\sum_{t=1}^{T} y_t \log(p_t) \tag{13}$$

where y_t and p_t respectively indicate the ground truth label and the predicted probability for training instance x_t.

To evaluate the noise filtering framework, I monitor the performance of multiple classifiers in an outcome classification task on both the flawed outcome annotations and the corrected outcome annotations.

Figure 3 illustrates the components of the proposed noise filtration framework combined with a classifier architecture used in evaluating the noise filtration process. The figure shows POS tags assigned to an input sentence in the second layer. Subsequently, a chunker that uses syntactic patterns in form of regular expressions identifies and eliminates unnecessary text following heuristics in Section 3.2.2.2 in the third layer. The corrected outcome span is then represented by a mean pool across GloVe initialised embeddings of its constituent words. I adopt 5 different architectures as classifiers to learn mapping each of the resulting outcome embeddings to the true outcome type label. These architectures include an LSTM [77], a CNN [109], a BiLSTM [244] as well as two bag-of-word models: SVM [171] and Multinomial Naive Bayes (MNB) [65]. Our evaluation protocol first performs supervised classification on both the initial extract of ca. 70,000 outcome spans from the original EBM-NLP and the corrected outcome spans that are ca. 32,000. I refer to the dataset with corrected outcome spans as EBM-NLP$_{\textbf{rev}}$. Using train/test splits that are respectively 80% and 20% for the noisy and corrected collections, I measure F1 performance (on test set) across the models in order to establish whether there is an increase/decrease/no change in performance after the noise filtration and whether it is consistent across all models.

Model	Adverse-effects [4489/1593]	Mental [8596/3875]	Mortality [1715/1176]	Pain [1649/839]	Physical [34997/18287]	Other [17996/5499]
Baseline (SVM)	0.55/0.65	0.63/0.72	0.62/0.89	0.70/0.77	0.67/0.85	**0.68/0.77**
CNN	0.31/0.44	0.58/0.69	0.49/0.61	0.52/0.70	0.55/0.71	0.57/0.59
LSTM	0.39/0.45	0.54/0.68	0.63/0.85	0.61/0.75	0.72/0.86	0.42/0.64
MNB	0.26/0.35	0.49/0.57	0.20/0.79	0.36/0.50	0.74/0.81	0.46/0.49
Bi-LSTM (BM)	**0.59/0.66**	**0.71/0.80**	**0.77/0.90**	**0.74/0.81**	**0.90/0.90**	0.62/0.75
BM - Flaw 1	0.37	0.69	0.83	0.65	0.78	0.58
BM - Flaw 2	**0.65**	**0.70**	**0.90**	**0.76**	0.85	**0.60**
BM - Flaw 3	0.56	0.70	0.72	0.66	**0.88**	0.59
BM - Flaw 4	0.51	0.63	0.50	0.70	**0.88**	0.57

Table 7: Average F1-score for each class before/after (before and after correcting outcome-spans). Best and second-best scores in bold and underlined respectively. Additional scores reported for the Best Model (BM) when subjected to data with flaws independently corrected. Enclosed in the brackets at the top is the instance count per class before/after, (Results rounded off to two decimal places).

3.2.3.1 Experiments and results

BiLSTM model outperforms all the other models in the outcome classification task. Additionally, there is a consistent increase in F1 in the evaluation performance on corrected outcome spans

I perform experiments (each using a different architecture defined above) aimed at evaluating how much the noise filtering framework (summarized in Algorithm 1 and illustrated in Figure 3) improves the F1 performance of the text classification task defined above. In all experiments, five-fold cross validation is used, with a batch-size of 500, trained for 100 epochs and a drop-out of 0.2 for each single fold.

Note: The bag-of-words models take as input, a Tf-Idf vector [243] representation of the words. The source code of our implementation of Algorithm Algorithm 1 and the classifiers built using tensorflow[5] is publicly availed.[6]

Results presented in Table 7 indicate that the accuracies of the classifiers (models) increased after correcting the errors in the outcome spans. Moreover, the increase was not only consistent across the five different models used, but even across prediction of the six classes/labels in the dataset. Notably, the BiLSTM outperforms all the other models, however, the bag-of-words SVM model is competitive i.e. it achieves the second highest F1 scores outperforming the other neural models. This suggests that despite the success of neural networks in language modelling tasks, some conventional learning algorithms like SVMs are highly effective in text classification tasks such as the classification of outcome spans in this evaluation.

5 https://www.tensorflow.org/
6 https://github.com/MichealAbaho/pico-outcome-prediction

Flaw	H1	H2	H3	H4	H5	H6	H7	H8	H9
Flaw 1	✓		✓						✓
Flaw 2		✓		✓	✓	✓			✓
Flaw 3							✓		✓
Flaw 4								✓	✓

Table 8: Each of the Flaws presented with the specific Heuristics used in correcting them. For example H1, H2 and H9 heuristics as mentioned in Section 3.2.2.2 are used to correct Flaw 1.

3.2.3.2 Flaw Analysis

In order to examine the impact the flaws individually had on the classification performance, the correction process was broken down to independently cater for the different flaws one by one. The best performing model, BiLSTM, would then be tested on input data where only annotations with flaw 1 were corrected (as shown in Table 8) and the rest ignored. This was repeatedly done for flaws 2, 3 and 4 as reported in the bottom half of Table 7. Flaw 5 was not considered in this additional analysis because of the extremely few cases it was responsible for.

Despite the largely analogous results, I observed that corrections targeted to fix flaw 2 alone, had a significantly higher impact on the performance, achieving higher F1-scores for the six classes with the exception of the Physical class. This implied that, granularity and distinctness is vitally important when automatically classifying not just outcomes but any relevant clinical entities in biomedical literature. Nonetheless, none of the F1-scores in this extended analysis would match up to the originally obtained F1-scores with all flaws corrected (line 5 - Table 7).

3.3 EBM-COMET: A NOVEL DATASET FOR OUTCOME DETECTION

EBM-COMET, a dataset that curates PubMed abstracts for OSD is introduced.

EBM has generally attracted less attention from the BioNLP community. This has mainly been attributed to the limited number of publicly available datasets with which to train and evaluate deep learning models [161]. More so, prior dataset construction efforts directed to EBM, have lacked a standard classification system to accurately inform their annotation process. Instead, they opted to use headings in structured abstracts (such as Participants, Interventions, Outcomes etc) as class labels [23, 84, 107]. Furthermore, I observed that majority of these efforts curated datasets for Sentence Level Classification (SLC) neglecting Token Level Classification (TLC) which would require granular (span of words) annotations. Nevertheless, Nye et al. [161] published EBM-NLP which contains granular annotations suited for

TLC, however, they also did not adopt any standard classification systems for the PICO elements [1, 2], but instead used arbitrary labels aligned to MeSH to annotate spans with information relevant to PICO. Moreover, EBM-NLP was discovered with flawed annotations which I addressed in the previous Section 3.2 on denoising crowdsourced annotations of outcome.

To further address the gaps mentioned in the preceding section, this chapter introduces EBM-COMET, a new dataset that curates PubMed [29] abstracts for EBM, particularly for Outcome span detection (OSD), a sequence labelling task to detect mentions of outcomes in clinical text. Similar to the previous Section 3.2, I adopt the widely acknowledged definition of an outcome as defined in Chapter 1 for the annotation process discussed in following sections. Different from all prior dataset construction efforts, EBM-COMET annotators use outcome domain classifications drawn from a recognised system i.e. an outcome taxonomy recently developed to standardise outcome reporting in electronic databases [52, 55] to annotate spans with outcomes. The taxonomy authors iteratively reviewed how core outcome sets (COS) studies within COMET database [89] categorised their outcomes. This review culminated into a taxonomy of 38 outcome domains hierarchically classified into 5 outcome types/core areas.

EBM-COMET was tested on all the other sub-tasks undertaken and discussed in the next three thesis chapters. Experiment results show that all evaluated models perform better on EBM-COMET, reaching an accuracy of 81.5% in the OSD task, compared to 53.1% on the EBM-NLP dataset. I however concentrate on the dataset construction process in this section and reserve the details on the evaluation on the dataset for the subsequent chapters.

A. Data collection

Using the Entrez API [185], I automatically fetch 300 abstracts from open access PubMed [29]. Our search criteria only retrieves articles of type "Randomised Controlled Trial". I relied on two domain-experts to review these abstracts and eliminate those reporting outcomes in animals (or non-humans). Each eliminated abstract was replaced by another reporting human outcomes from PubMed.

B. Annotation

The two experts I work with have sufficient experience in reviewing human health outcomes in clinical trials. Some of their work pertaining to outcomes in clinical trials includes [60, 111, 225, 226]. These experts jointly annotate granular outcomes within the gathered abstracts resulting into EBM-COMET using guidelines below. We are aware of annotation tools such as BRAT [196], however because of the nature of the annotations i.e. some with contiguous outcome spans,

the experts prefer to directly annotate them in Microsoft text documents.

Core area	Outcome domain	Domain symbol	Explanation
Physiological	Physiological/Clinical	P 0	Includes measures of physiological function, signs and symptoms, laboratory (and other scientific) measures relating to physiology.
Death	Mortality/survival	P 1	Includes overall (all-cause) survival/mortality and cause-specific survival/mortality, as well as composite survival outcomes that include death (e.g. disease-free survival, progression-free survival, amputation-free survival).
Life impact	Physical functioning	P 25	Impact of disease/condition on physical activities of daily living (for example, ability to walk, independence, self-care, performance status, disability index, motor skills, sexual dysfunction. health behaviour and management).
	Social functioning	P 26	Impact of disease/condition on social functioning (e.g. ability to socialise, behaviour within society, communication, companionship, psychosocial development, aggression, recidivism, participation).
	Role functioning	P 27	Impact of disease/condition on role (e.g. ability to care for children, work status).
	Emotional functioning/wellbeing	P 28	Impact of disease/condition on emotions or overall wellbeing (e.g. ability to cope, worry, frustration, confidence, perceptions regarding body image and appearance, psychological status, stigma, life satisfaction, meaning and purpose, positive affect, self-esteem, self-perception and self-efficacy).
	Cognitive functioning	P 29	Impact of disease/condition on cognitive function (e.g. memory lapse, lack of concentration, attention); outcomes relating to knowledge, attitudes and beliefs (e.g. learning and applying knowledge, spiritual beliefs, health beliefs/knowledge).
	Global quality of life	P 30	Includes only implicit composite outcomes measuring global quality of life.
	Perceived health status	P 31	Subjective ratings by the affected individual of their relative level of health.
			Includes outcomes relating to the delivery of care, including - adherence/compliance, withdrawal from intervention e.g. time to treatment failure). - tolerability/acceptability of intervention. - appropriateness, accessibility, quality and adequacy of intervention. - patient preference, patient/carer satisfaction (emotional rather than financial burden). - process, implementation and service outcomes (e.g. overall health system performance and the impact of service provision on the users of services).
	Personal circumstances	P 33	Includes outcomes relating to patient's finances, home and environment.
Resource use	Economic	P 34	Includes general outcomes (e.g. cost, resource use) not captured within other specific resource use domains.
	Hospital	P 35	Includes outcomes relating to inpatient or day care hospital care (e.g. duration of hospital stays, admission to ICU).
	Need for further intervention	P 36	Includes outcomes relating to, - medication (e.g. concomitant medications, pain relief) - surgery (e.g. caesarean delivery, time to transplantation) - other procedures (e.g. dialysis-free survival, mode of delivery)
	Societal/carer burden	P 37	Includes outcomes relating to financial or time implications on carer or society as a whole e.g. need for home help, entry to institutional care, effect on family income
Adverse events	Adverse events/effects	P 38	Includes outcomes broadly labelled as some form of unintended consequence of the intervention e.g. adverse events/effects, adverse reactions, safety, harm, negative effects, toxicity, complications, sequelae. Specifically named adverse events should be classified within the appropriate taxonomy domain above

Table 9: A taxonomy of outcome classifications developed and used by Dodd et al. [55] to classify clinical outcomes extracted from biomedical articles published in repositories that include Core Outcome Measures in Effectiveness Trials (COMET), Cochrane reviews and clinical trial registry

Annotation category	Annotated text	Outcome span	Outcome domain
Simple	... Peer support education also benefited the <**P 0**> blood glucose control </> in the general population ...	blood glucose control	• Physiological
	... Tai Chi may alleviate <**P 0, 28**> depression </> of the elderly through modulating autonomous nervous system or <**P 0**> heart rate variability </> parameters ...	depression	• Physiological • Emotional functioning
		heart rate variability	• Physiological
Complex	... The objective of this study was to evaluate <**P 0**>(S2) right heart size and <**P 0**>function </> assessed by echocardiography during long term treatment with riociguat ...	right heart size	• Physiological
		right heart function	• Physiological
	... Their relationship to <**P 29**>(E1) Neurological and <**P 29**> Cognitive functions </> in PKU Patients ...	Neurological functions	• Cognitive functioning
		Cognitive functions	• Cognitive functioning

Figure 4: Sample annotations of outcomes depicting the annotation style with each example showing the outcome span and its assigned outcome domain label.

B.1 Annotation guidelines

The annotators are tasked to identify and verify outcome spans and then assign each an outcome domain referenced from the taxonomy presented in Table 9. The annotators are instructed to assign each span all relevant outcome domains.

B.2 Annotation heuristics

Annotation heuristics used to guide the annotation process. Demarcating spans of outcomes using opening and clsong tags with the outcome domain positioned besides the opening tag.

For annotation purposes, I firstly assign a unique symbol to each outcome domain (drawn from domain symbol column in Table 9). The annotators are then instructed to use these symbols to label the outcome spans they identify. Annotation using these symbols rather than the long domain names is less tedious. Furthermore, I instruct annotators to use xml tags to demarcate the spans, such that an identified span is enclosed within an opening tag with the assigned domain symbol and a closing tag. I refer to easily identifiable outcome spans as simple annotations, and the more difficult ones requiring more demarcation indicators as complex annotations. Figure 4 shows examples of the annotations described below,

1. Simple annotations
    1.1. <P XX>...</>: Indicates an outcome belongs to domain XX (where XX can be located in the taxonomy 9).
    1.2. <P XX, YY>...</>: Indicates an outcome belongs to both domains XX and YY.
2. Complex annotations
   Some spans are contiguous in such a way that, they share a word or words with other spans. For example, two outcomes

can easily be annotated as a single outcome because they are conjoined by a dependency word or punctuation such as "and", "or" and commas. I am however fully aware, that this contiguity previously resulted in multiple outcomes annotated as a single outcome in previous datasets [1]. Therefore, annotators are asked to distinctively annotate them as below,

2.1. Contiguous spans sharing bordering term/s appearing at the start of an outcome span should be annotated as follows,
<P XX>(S#)...<P XX>...</>: which indicates that, two outcomes are belonging to domain XX that share # of words at the start of the annotated outcome span.

2.2. Contiguous spans sharing bordering term/s appearing at the end of an outcome span, should be annotated as follows,
<P XX>(E#)...<P XX>...</>: The opposite of the notation above indicating that, two outcomes are belonging to domain XX that share # of words at the end of the annotated outcome span.

Simple annotations are those with the template <P domains>[outcome span</>] whereas complex annotations have a few more identifiers to indicate contiguity e.g. <P domains>[S#][outcome span</>, where S# implies shares # tokens at start of span.

B.3 Annotation consistency and quality

In the last phase of the annotation process, the annotations are extracted into a structured format (excel sheet) for the annotators to review them, make necessary alterations based on their expertise judgement as well as handle minor errors (such as wrong opening or closing braces) that result from the manual annotation processes. I do not report inter-annotator agreement because the two annotators did not conduct the process independently, but rather jointly. Having previously worked together on similar annotation tasks, they hardly disagreed but whenever either was uncertain or disagreed, they discussed between themselves and concluded.

3.4 LABEL DENOISING USING COMPARABLE DATASETS

As a final step in achieving this thesis's first objective which aims to evaluate and improve the reliability of current outcome annotations in weakly labelled datasets, This section attempts to denoise the arbitrary outcome classifications (labels) in EBM-NLP by aligning them to standard outcome classifications proposed by Dodd et al. [55] and used to annotate EBM-COMET. These standard classifications were found (after extensive analysis and testing) to provide sufficient granularity and scope of trial outcomes.

I introduce a flexible, re-usable unsupervised text alignment approach that extracts parallel annotations from comparable datasets. I use this alignment for data augmentation in a low-resource setting

in a proposal made to jointly detect outcomes (i.e OSD) and classify outcomes (OC) in Chapter 5.

3.4.1 Label alignment (LA) task definition

I introduce a an unsupervised text alignment approach that aligns arbitrary labels to standardised labels. An arbitrary label is changed to the standardised label its most similar or aligned to.

Given two datasets $\mathcal{S}$ and $\mathcal{T}$ with comparable content, with $\mathcal{S}$ containing x labels such that $L_s = \{l_s^1, \ldots, l_s^x\}$ and $\mathcal{T}$ containing y labels $L_t = \{l_t^1, \ldots, l_t^y\}$, I design LA to measure the similarity between each pair of labels (l_s, l_t). For this purpose, I first create an embedding for each label l_s in a sentence $s(\in \mathcal{S})$ by applying mean pooling over the span of embeddings (extracted using pre-trained BioBERT [121]) for the tokens corresponding to an outcome annotated with l_s as shown in (14). Next, I average the embeddings of all outcome spans that are annotated with l_s in all sentences in $\mathcal{S}$ to generate an outcome type label embedding $\mathbf{l}_s$ as shown in (15). Likewise, I create an outcome type label embedding, $\mathbf{l}_t$ for each outcome type in the target dataset $\mathcal{T}$. After generating label embeddings for all outcome types in both $\mathcal{S}$ and $\mathcal{T}$, I compute the cosine similarity between each pair of $\mathbf{l}_s$ and $\mathbf{l}_t$ as the alignment score between each pair of labels l_s and l_t respectively.

$$\mathbf{O}_{l_s} = \frac{1}{d} \sum_{i}^{i+(d-1)} \text{Biobert}(w_i) \tag{14}$$

where O_{l_s}, is an outcome span annotated with outcome type label l_s, i and $i + (d - 1)$ are the locations of the first and last words of the outcome span.

$$\mathbf{l}_s = \frac{1}{|l_s|} \sum_{1}^{|l_s|} \mathbf{O}_{l_s} \tag{15}$$

where $|l_s|$ is the number of outcome spans annotated with label l_s and $\mathbf{l}_s$ is label l_s embedding.

3.4.2 Evaluation experiments and results

Table 10 shows the similarity scores for label pairs (l_s, l_t) across $\mathcal{S}$ (EBM-COMET) and $\mathcal{T}$ (EBM-NLP) respectively. For each label (which is an outcome domain) in EBM-COMET, I identify the EBM-NLP label which is most similar to it by searching for the least cosine distance across the entire column. After identifying those pairs that are most similar, I automatically replace outcome type labels in EBM-NLP with EBM-COMET outcome type labels as informed by the similarity measure.

Results show that Physiological outcomes (containing domain P o) are similar to Physical outcomes and therefore the latter outcomes

	Physiological	Mortality	Life-Impact									Resource-use			Adverse-effects
	P 0	P 1	P 25	P 26	P 27	P 28	P 29	P 30	P 31	P 32	P 33	P 34	P 35	P 36	P 38
Adverse-effects	0.0615	0.1532	0.1226	0.1893	0.2001	0.1348	0.1169	0.2555	0.2320	0.0897	0.1936	0.2561	0.1768	0.1043	**0.0562**
Mental	0.0387	0.1829	**0.0444**	**0.0928**	**0.1529**	**0.0623**	**0.0419**	**0.2214**	**0.1624**	**0.0624**	**0.1063**	**0.2537**	0.1955	0.1041	0.1904
Mortality	01330	**0.0187**	0.1722	0.2562	0.2563	0.2171	0.1821	0.2594	0.2956	0.1559	0.2349	0.2855	0.1976	0.1905	0.2082
Pain	0.0947	0.2310	0.1266	0.2181	0.1906	0.1316	0.1634	0.2662	0.2089	0.1290	0.2209	0.2770	0.2269	0.1422	0.2096
Physical	**0.0114**	0.1582	0.0698	0.1494	0.1878	0.1126	0.0788	0.2363	0.2059	0.0639	0.1461	0.2539	**0.1758**	**0.0761**	0.1803

Table 10: Cosine distance between representations of EBM-NLP labels (first column) and EBM-COMET labels (top and second row). EBM-COMET outcome type labels were drawn from the outcome domains defined in [55] taxonomy. Due to space limitations, I denote these domains as P X such as P 0, P 1 etc. The taxonomy hierarchically categorised them into 5 outcome types which are accordingly included in the top row. Outcome domains definitions are, P 0-Physiological/clinical, P 1-Mortality/survival, P 25-Physical functioning, P 26-Social functioning, P 27-Role functioning, P 28-Emotional functioning/wellbeing, P 29-Cognitive functioning, P 30-Global quality of life, P 31-Perceived health status, P 32-Delivery of care, P 33-Personal circumstances, P 34-Economic, P 35-Hospital, P 36-Need for further intervention, P 37-Societal/carer burden, P 38-Adverse events/effects.

are labelled Physiological, Life-Impact outcomes are similar to Mental outcomes and therefore the latter outcomes are labelled Life-Impact. Mortality and Adverse-effects outcomes both remain unchanged because both categories exists in source and target datasets, and their respective outcomes are discovered to be similar. I evaluate the joint learning architecture I propose in Chapter 5 on the resulting merged dataset, and additionally, evaluate the alignment approach by comparing the performances before and after merging. Overall, an average improvement of 2.5% and 5.5% in F1 for the OSD and OC tasks across both the EBM-COMET and EBM-NLP datasets test sets was observed. Full details on this evaluation are deferred to Table 5.4.

3.5 DISCUSSION AND SUMMARY

To tackle the challenges that motivated the first objective of this thesis that aims to improve the reliability of outcome annotations, three different measures are undertaken and exhaustively discussed in this chapter. The chapter begun by proposing a hybrid noise filtering framework that combines POS tagging and rule-based chunking to denoise flawed outcome annotation spans in a crowdsourced EBM benchmark dataset (EBM-NLP). The framework uses a collection of heuristics that use lexical and syntactic information to filter out noise from annotated data. Each heuristic is strategically created to filter out specific noise (flaw), however correction of each flaw is not necessarily limited to a single heuristic. Experiments targeting a task to classify an outcome span (OC) showed that the proposed framework led to an improvement in the F1 classification scores for each outcome type.

The second measure introduced EBM-COMET, a dataset of PubMed abstracts in which outcome spans are expertly annotated. This dataset is distinct from earlier efforts because it uses standardised outcome classification labels drawn from a recently proposed taxonomy of standardised outcome classifications [55]. EBM-COMET is built to support BioNLP tasks aimed at EBM, particularly for OSD and OC tasks. In several experiments, I observe PLMs fine-tuned on EBM-COMET consistently produce better F1 scores than those fine-tuned on EBM-NLP in both OSD and OC. Discussions on the impact EBM-COMET makes, are however reserved for chapters 4, 5 and 6 presenting other methods that this thesis proposes.

Finally, the chapter proposes a label denoising approach that aims to automatically correct weak labels in the EBM-NLP corpus by replacing them with more informative and standardised outcome classification labels drawn from the outcome taxonomy proposed by Dodd et al. [55]. The denoiser is a flexible, re-usable unsupervised text alignment approach which extracts parallel annotations from comparable datasets, where one of the datasets is considered to have the standardised target labels. The intuition behind this denoiser, is to determine annotations in the dataset (with standardised labels) that a weak annotation is most similar too, and subsequently automatically re-annotate the weak one with the similar standardised label. I use this alignment for data augmentation in a low-resource setting in a proposal made to joint OSD and OC in Chapter 5.

Using the newly introduced EBM-COMET dataset and the revised version of the EBM-NLP dataset, the next chapter assesses the performance of various contextualised embeddings models such as BERT and ELMo in the OSD task.

4

ASSESSMENT OF CONTEXTUALISED REPRESENTATIONS IN DETECTING OUTCOMES

4.1 INTRODUCTION

Encoding surrounding context of a pivot word to produce a contextualised (or context-dependent) word representation (embedding), has been largely responsible for the recent upheaval in NLP [54, 136, 167, 235]. While prior generic embedding models provide a single static vector for a word regardless of the context around it [149, 166], Contextualised Language Models (CLMs) provide a different vector for each word depending on the context in which it is mentioned. This vector variation is intentional simply because, different contexts will often trigger different meanings of a word, which can invariably be relevant in disambiguation for language [138]. This context-encoding ability has been the distinguishable trait behind the superior performance that Contextualised Representations (CRS) (provided by CLMs) have achieved in a broad range of NLP downstream tasks like NER [207], QA [175], Machine Translation (MT) [228], NLI [47] etc.

This context-encoding ability has been the distinguishable trait behind the superior performance that CRS (provided by CLMs) have achieved in a broad range of NLP downstream tasks like NER, QA, MT etc

Following their success in generic-domain tasks, CRS have further led to impressive gains in many domain-specific tasks [13, 32, 121]. The notion generic-domain, is used to denote a distribution over a language characterising a diverse set of topics, whereas domain-specific is used to denote a distribution over a language characterising a single topic [72]. Focused on the biomedical domain, CRS have improved performance in automatic recognition of diseases and chemicals [154], gene-disease interactions [121], drug-drug interactions [154], chemical-protein interactions [72], clinical NLI tasks [148] etc. Unlike these BioNLP tasks, CRS have been underutilised for OD tasks earlier defined in Chapter 1 i.e. Outcome Span Detection (OSD) and Outcome Classification (OC). In as such, there is still little knowledge about the capability and limitations of CRS in encoding and detecting health outcomes from clinical text.

This chapter carries out a comprehensive analysis to investigate the performance of biomedical CRS in tasks such as OSD using the denoised and newly introduced datasets in the preceding chapter (Chapter 3). The goal in the OSD task is to detect and extract outcome spans from clinical text. For example, in a sentence, "The patient's ***systolic blood pressure*** rose over the course of treatment.", OSD extracts all outcome spans such as those underlined and in bold font. This enables those searching the literature including patients and policymakers to identify research that addresses the health outcomes of

most importance to them [19]. Following previous studies that investigated which CRS or embeddings are best suited for clinical-NLP text classification tasks [145], this work is focused on probing for some consensus amongst various SOTA domain-specific CRS, determining which CRS are best suited for OSD.

Feature extraction involves adapting frozen model parameters in a down stream model whereas, Fine-tuning involves continually training a models parameters however using a downstream task.

Specifically, the chapter scrutinizes two pre-trained model adaption (TL) paradigms of fine-tuning and feature extraction [169]. With feature extraction, model parameters are frozen and used in a down stream model whereas, in fine-tuning, a model continues to train its parameters however using a downstream task. In summary, the chapter: (1) performs an in-depth comparison between fine-tuning CRS and adapting frozen CRS in a feature extraction approach, (2) performs a qualitative assessment of accurately detecting full mentions of outcome spans i.e. full outcome span evaluation strictly rewards models for correctly detecting both the entity-span words and the entity classification label, unlike some traditional sequence labelling evaluation which credits models for detecting a correct entity classification label regardless of a partial match or overlapping entity-span boundaries or for the exact entity-span boundaries regardless of the classification label [39, 69], (3) compares the performance of the CRS in our experimental setup to the leader-board[1] performance on extracting PICO elements from the original EBM-NLP dataset [161].

The remainder of the chapter begins with a discussion of biomedical CLMs used in Section 4.2. This is followed by Section 4.3 which describes an architecture that I use in adapting the CLMs. Within Section 4.3 is a discussion on fine-tuning and feature-based adaption approaches, where the latter involves systematically building a neural model tailored for OSD tasks. A discussion on the evaluation of the above mentioned models in the OSD task highlighting the contributions in the preceding paragraph is provided in Section 4.4 before summarizing the chapters main findings in Section 4.5.

4.2 BIOMEDICAL CONTEXTUAL LANGUAGE MODELS

CLMs built by pre-training on heterogeneous or general corpora are called generic-domain CLMs, whereas those that are built by pre-training on domain-specific corpora are called domain-specific CLMs. Intuitively, pre-training CLMs on biomedical text produces biomedical CLMs. As earlier discussed in the preamble to this chapter, CRS provided by these pre-trained models have elevated the performance in several downstream general and domain-specific NLP tasks.

While there are a few works that pre-train models on domain-specific corpora (i.e. domain adaptive pre-training [DAPT; 72]) from scratch [86, 236], majority of pre-trained biomedical CLMs typically follow a standard pre-training approach that involves initialising vanilla

1 https://ebm-nlp.herokuapp.com/

CLMs such as BERT [54] and then continue the pre-training process using biomedical text such as PubMed articles [72]. Common pre-training architectures typically include the Masked Language Model (MLM) architecture [10, 18, 121], forward and backward LSTMs [97, 189], both of which are further discussed under the individual CLMs presented below.

I compare a set of 5 biomedical CLMs derived by pre-training 3 main CLM architectures described below.

1. BERT [54] : a CLM built by learning deep bidirectional representations of input words by jointly incorporating left and right context in all its layers. BERT is pre-trained on 2.5M words from Wikipedia and 0.8M words from the BookCorpus [248] using two unsupervised tasks. The first being the MLM task, which masks a portion of the input words and trains models to predict the original value of masked words in an input sentence. BERT maximizes the log likelihood of a word encoded using a self-attention mechanism [213] which incorporates information about words around it within a given input sentence (as shown in (16)). The second task is Next sentence prediction (NSP), in which the model receives a pair of sentences with one following or subsequent to another. In the NSP task, BERT learns to predict if indeed the subsequent sentence comes after the first sentence.

$$\sum_{k=1}^{N} p_\theta(x_k|x_k,\ldots,x_{k-1},x_{k+1},\ldots,x_N) \tag{16}$$

BERT, ELMo and FLAIR produce bidirectional representations by conditioning on the left and right context.

2. ELMO [167] : a CLM that learns deep bidirectional representations of input words by jointly maximizing the probability of forward and backward directions in a sentence as shown in (17). ELMo computes a weighted sum of the hidden states from each layer of a 2-layered BiLSTM to obtain a word embedding. ELMo was originally pre-trained on approximately 30M sentences from new stories in a monolingual corpora [34].

$$\sum_{k=1}^{N} (\log p_\theta(x_k|x_1,\ldots,x_{k-1}) + \log p_\theta(x_k|x_{k+1},\ldots,x_N)) \tag{17}$$

3. FLAIR [7], a character-level CLM which learns representations of each character by incorporating character information around it within a sequence of words. Similar to ELMo, FLAIR, uses both forward and backward LSTMs, however, instead of computing a task specific weighting, FLAIR concatenates representations across all layers for each character. FLAIR was originally pre-trained with 1B news word corpus [34].

CLM	Biomedical Variant	Pre-trained on
BERT	BioBERT	4.5B words from PubMed abstracts + 13.5B words from PubMed Central (PMC) articles.
	SciBERT	1.14M Semantic scholar papers [43] (18% from Computer science and 82% from biomedical domains).
	ClinicalBERT	2 million notes in the MIMIC-III v1.4 database [44] (hospital care data recorded by nurses). (Bio+Clinical BERT is BioBERT pre-trained on the above notes)
	DischargeSummaryBERT	Similar to ClinicalBERT but only discharge summaries are used (Bio+DischargeSummary BERT is BioBERT pre-trained on the summaries)
ELMo	BioELMo	10M PubMed abstracts (ca. 2.64B tokens)
FLAIR	BioFLAIR	1.8m PubMed abstracts.

Table 11: A catalogue of CLMs evaluated on EBM datasets to assess their capability in Outcome Span Detection (OSD) and Outcome Classification (OC).

Table 11 shows corpora used in pre-training the biomedical CLMs used in the assessment approach. It is important to note that, I chose these CLMs because at the point of conducting the analysis covered in this chapter, these particular CLMs had been extensively used in various BioNLP tasks. A few other biomedical CLMs such as UmlsBERT [148], Biomed_RoBERTA [72] have since been released and I use them in work discussed in the next chapters.

4.3 ADAPTING PRE-TRAINED BIOMEDICAL LANGUAGE MODELS TO OSD

OSD aims to extract outcome span or spans within a sentence drawn from an RCT abstract.

Inspired by Nye et al. [161] who propose a new corpus to support building NLP applications to address more complex tasks in EBM, such as granular outcome classification and extraction of codified types, I define OSD as a sequence labelling task as follows,

Given a sentence s of n words, $s = w_1, \ldots, w_n$ within a RCT abstract, OSD aims to extract an outcome span $o = w_x, \ldots, w_d$ within s, where $1 \leqslant x \leqslant d \leqslant n$. In order to extract outcome spans such as o, each word is labelled using the "BIO" tagging scheme [182] where "B" denotes the beginning word of the outcome span, "I" denotes inside the outcome span or words following the beginning word and "O" denotes all non-outcome words. Some abstract sentences have got multiple outcome spans as shown in the example in Figure 5,

In this adaptation approach, I design two setups.

***Fine-tuning*:** In this setup, inputs and outputs of the OSD task (defined above) are plugged into pre-trained biomedical CLMs listed in Table 11. The CLMs are then further trained using the outcome datasets EBM-COMET and revised version of the EBM-NLP (denoted as EBM-NLP$_{\textbf{rev}}$ in the rest of this chapter), both of which were intro-

Among patients who received sorafenib, the most frequently reported , adverse events (o_1) were grade 1 or 2 events of rash (o_2) (73%), fattigue (o_3) 67%, hypertension (o_4) (55%) and diarrhea (o_5) (51%).

Figure 5: An example RCT abstract sentence with outcome spans that OSD aims to extract.

duced in Chapter 3. The goal of this process is to update or adjust the pre-trained CLMs parameters to the objectives of the OSD task expounded on in Section 4.3.1.

***Feature extraction*:** This setup uses the pre-trained biomedical CLMs listed in Table 11 as feature extractors i.e. the CRS of the words are extracted from the last layers of the models. In this approach, I build a custom neural language model to train these extracted contextualised features (also known as frozen embeddings) on the OSD task using the same datasets as those in the *fine-tuning* setup. Besides the contextualised features, this setup is further used to assess non-contextual (context-independent) feature embeddings such as word2vec [149].

The prevailing variation in fine-tuning approaches makes it difficult to fully understand the impact of PLMs.

4.3.1 Fine-tuning based adaptation

Various task-specific model objectives and fine-tuning methods are adopted when performing task-specific fine-tuning in NLP. Whereas this prevailing variation in fine-tuning approaches is not completely surprising since different tasks seek different outputs, it makes it difficult to fully understand the impact different pre-trained CLMs have on a specific task. To facilitate a head-to-head comparison between

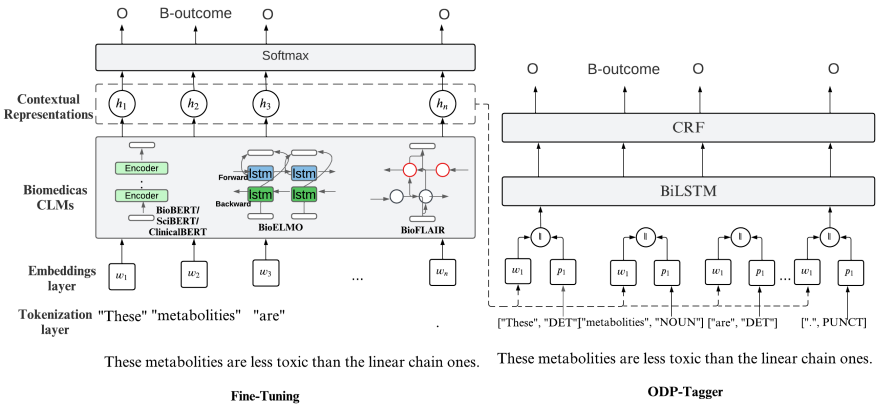


Figure 6: OSD for for the two assessment setups, Fine-tuning and Feature extraction using the ODP tagger. Contextual representations extracted from the the Biomedical CLMs is fed into the downstream ODP-tagger model. In addition to these, I feed POS embeddings corresponding to the POS tag for each tokenized word.

the different CLMs in my primary investigation, I fix a task-specific layer on top of the biomedical CLMs generating CRS to predict the output of the OSD task.

Figure 6 shows the architecture used in fine-tuning the CLMs. An input sentence extracted from an RCT abstract is pre-processed by splitting it into a sequence of tokens or characters (**tokenization layer**). I use the tokenization algorithms that were originally used in the work proposing the CLMs. For BERT, the WordPiece algorithm which effectively splits rare words into sub-words and leaves the frequently used words in the vocabulary unsplit [228] is used. The intuition is that, this will help the model to learn that most rare words are formed by joining the frequent words, e.g. "tokenization" can be split into "token" and "##ization", where the "##" in the second sub-token is to imply that "##ization" is a piece of a word rather than a word itself. Inaddition to the sentence tokens, two extra tokens [CLS] and [SEP] are appended to the beginning and the end of the token sequence.

BERT uses the WordPiece algorithm, ELMo uses the Moses Tokenizer, whereas FLAIR uses character tokenization.

The tokenized input is then embedded using BERT's **embedding layer**. This layer sums up the WordPiece-generated embeddings [228], segment embeddings (which are used to distinguish between input sentences especially if there is multiple sentences (in my case, it's a single sentence)) and position embeddings (which preserve the information about the position of the words in the input sentence). These embeddings are encoded to generate a hidden state as a contextual representation (**Biomedical CLMs layer**) using 12 Transformer architectures stacked together which, each include a self-attention layer and feed-forward neural networks [213]. I use the BERT_{base} which contains 12 layers (Transformer architectures) to generate CRS (**contextual representation layer**) for each token as shown in (18).

With ELMo, the input sentence is tokenized using the Moses tokenizer which separates punctuation from words however preserving special tokens such as URLs and dates [112]. Max-pooling over character-level CNN encodings is used to generate embeddings that ELMo treats as token embeddings. These embeddings are encoded by a 2-layered BiLSTM, and a concatenation of the BiLSTM internal states is used to generate contextual representations as the hidden states as shown in (18).

Finally with FLAIR, the input sentence is split into individual characters. FLAIR encodes input characters of each word using 1-layer of forward and backward LSTMs (fLM and bLM respectively). To extract a hidden state for each word, the last and first character hidden states extracted from the fLM and bLM respectively are concatenated to generate contextual representations as shown in (18).

$$\mathbf{h}_i = \text{CLM}(w_i) \tag{18}$$

where w_i is the token embedding corresponding to the word at position i, CLM $\in$ {BERT-variants, BioELMo, BioFLAIR}.

A softmax function is then applied to return a probability of each label for each position in the sentence s using (19),

$$y = \text{softmax}(\mathbf{W} \cdot \mathbf{h}_i + b) \tag{19}$$

where $\mathbf{W} \in \mathbb{R}^{|\mathcal{L}| \times k}$ i.e. $\mathbf{W}$ is a matrix with dimensions $|\mathcal{L}|$ (size of label set) $\times k$ (hidden-state size). $\mathcal{L}$ represents the set of outcome type target labels. Given the probability distribution the softmax generates at each position, $\underset{\theta}{\text{argmax}} P(y|\boldsymbol{w}_i; \theta)$ is used to return the predicted outcome type label.

4.3.2 Feature-extraction (feature based) adaptation

To transfer the CRS derived from the biomedical CLMs, I introduce an OSD custom module that called an Outcome Detection tagger (ODP-tagger). ODP-tagger is built by augmenting a BiLSTM with in-domain resources including clinically oriented POS and PubMed word2vec embeddings [149]. The architecture of the tagger is composed of four layers as shown in Figure 6: *tokenization layer*, *embedding layer*, *BiLSTM (encoder) layer* and a *CRF (classification) layer*. Pertinent details of the four layers are described below.

Tokenization layer: Splits an input sentence into tokens and additionally outputs a corresponding POS term for each token. A POS feature is added in order to enrich each tokens representation with POS information in an approach similar to how prior neural classifiers are enhanced with character and n-gram features [133].

Embedding layer: Takes as input, a sequence of token embeddings which are CRS extracted from the biomedical CLMs (demonstrated by the dotted line from Fine-tuning to Feature extraction). Prior works on probing CLMs have shown that performance of pre-trained features can vary depending on which layer the pre-trained features are extracted from [74, 134]. Majority of these works demonstrate that intermediate layers are more transferable than upper layers and that the upper layers are more discriminative than lower and intermediate layers for classification tasks [74, 134]. I however do not further probe layer-wise performance especially given that BioELMo and BioFLAIR have an extremely small number of layers compared to BERT-variants, but instead extract representations from the last layer for each of the respective CLM encoders in the head-to-head comparison.

Prior works on probing CLMs indicate that Intermediate layers are more transferable than upper layers and that the upper layers are more discriminative than lower and intermediate layers for classification tasks.

In addition to the token contextual embeddings, I randomly initialize POS embeddings for POS terms that are obtained using a trained Stanford POS tagger that is systematically selected as discussed in Section 4.3.2.1. Each POS embedding corresponds to a POS term assigned to a token. The token is therefore represented by concatenating a contextual embedding $\boldsymbol{w}$ and a POS embedding $\mathbf{p}$. Besides the contextual embeddings, I also test non-contextual embeddings (such as

word2vec) for each token as discussed in Section 4.3. I train word2vec (W2V) on 5.5B tokens of PubMed and PubMed Central (PMC) abstracts to obtain non-contextual token embeddings.

BiLSTM (encoding) layer: The token embeddings are then encoded by a 2-layered BiLSTM to obtain hidden-states for each sequence position as shown in (20),

$$\mathbf{h}_i = \alpha(\mathbf{W}[\boldsymbol{w}_i; \mathbf{p}_i] + b) \tag{20}$$

where $\boldsymbol{w}_i \in \mathbf{E}^w$ and $\mathbf{p}_i \in \mathbf{E}^p$, $\{\mathbf{E}^w, \mathbf{E}^p\} \in \mathbb{R}^{n \times d}$ denote Word and POS matrices, each containing d-dimensional embeddings for n words and n corresponding POS terms. $\boldsymbol{w}_i$ and $\mathbf{p}_i$ are the word and POS embeddings representing the i^{th} word and its POS term respectively, ; implies a concatenation operation and then α is a linear activation function that generates hidden states for the input words.

CRF (classification) layer: A CRF layer is used for classification given the hidden state $\mathbf{h}_i$. A CRF is an undirected graphical model which defines a conditional probability distribution over possible labels [116]. The layer outputs the predicted label or class for each token.

In order to select the final components used in the ODP-tagger, I conduct investigative experiments to monitor the performance improvements brought about by each of the components as detailed in the following sections. For these experiments, the EBM-NLP$_{\mathbf{rev}}$ dataset, (which has outcome spans annotated with outcome types) and a BiLSTM network are used. Sections Section 4.3.2.1 to Section 4.3.2.3 includes further details on this investigation.

4.3.2.1 Probing for biomedical POS tagger

I compare the performance of 3 POS taggers, which include, taggers of 2 popular, fully established NLP libraries in spaCy[2] and Stanford CoreNLP[3], and a tagger specifically tuned for POS tagging tasks on biomedical text known as the Genia-Tagger [209]. The Genia-Tagger is pre-trained on a collection of articles extracted from the MEDLINE database [162]. To avoid any biased analysis in the comparative study, I customise spaCy and Stanford CoreNLP taggers for biomedical text tagging by training them on a corpus of 6,700 Medline sentences (MedPOST) annotated with 60 POS tags [192]. These 3 taggers are each used to provide POS features to input samples (outcomes spans) in an OC task which classifies the samples into outcome types predefined in EBM-NLP$_{\mathbf{rev}}$ dataset, these include Physical, Pain, Mental, Mortality, Adverse effects and Other. A BiLSTM network and a softmax classification layer are used to complete the OC task using the EBM-NLP$_{\mathbf{rev}}$ dataset that contains 40092 sentences split into train (80%)

2 https://spacy.io/
3 https://nlp.stanford.edu/software/tagger.html

and test (20%) sets. In the evaluation phase on the test set, I observe the model using trained Stanford CoreNLP tagger outperforming the other models (using the other two taggers) as shown in Table 12 results. I therefore use Stanford CoreNLP tagger for POS tagging in the OSD task in ODP-tagger's tokenization layer.

Stanford CoreNLP tagger emerges the suitable atgger to use in providing POS feature in ODP-tagger

	EBM-NLP$_{rev}$
BiLSTM-spaCY-MedPOST	80.5
BiLSTM-stanford-MedPOST	81.3
BiLSTM-Genia-Tagger	79.0

Table 12: Macro-average F1 percentage scores in the OC task on EBM-NLP$_{rev}$ corpus. Biomedical POS taggers including spaCY-MedPOST, stanford-MedPOST and Genia-Tagger are used to provide POS features which alongside the text are used in training the BiLSTM model.

	Adverse-effects	Mental	Mortality	Pain	Physical	Other
# of samples	1593	3875	1176	839	18287	5499

Table 13: Frequency distribution of samples in across outcome types or labels in EBM-NLP$_{rev}$

4.3.2.2 Probing for a loss function

During an initial exploration of the EBM-NLP$_{rev}$ dataset, I observed a huge imbalance in the label (outcome type) distribution across the dataset. As shown in Table 13, the Physical outcome type is dominant with a significant number of samples while the other outcome types are represented by relatively fewer samples. To address the problems that arise from imbalanced data, I test 3 cost-sensitive functions in the ODP-tagger framework premised on a log-likelihood objective $\log p(y|w)$, (log probability of label y given input word w) in order to identify a suitable learning loss for the OSD task. In the OSD task, all models are each trained to maximize the probability of the labels given each word $w_i \in s$ as shown in (21).

$$\underset{\theta}{\operatorname{argmax}} P(y_n|\boldsymbol{w}_n;\theta) \tag{21}$$

The training loss is defined as a cross entropy loss given by (22).

$$\text{ODP}_{\text{loss}} = -\sum_{(S,L)\in\mathcal{T}} \sum_{i}^{n} p(y|\boldsymbol{w}_i) \tag{22}$$

where $\mathcal{T}$ is the training set containing sentences, $\boldsymbol{w}_i \in S$ and $y \in L$.

IMPUTED INVERSE LOSS (IIL): Empirically sets each label's (outcome type's) weights to be inversely proportional to the label frequency [82] shown in Table 13 using a scaling factor β.

$$\text{IIL} = \beta \cdot \text{ODP}_{\text{loss}} \tag{23}$$

Two variants of the scaling factor β in the IIL equation (23) are checked. IIL_1 where $\beta = \frac{1}{N_y}$ and a smoothed version IIL_2 where $\beta = \frac{1}{\sqrt{N_y}}$, where N_y is the number of training samples labelled y or frequency of ground truth label y.

CLASS BALANCED LOSS (CB): CB introduces a weighting factor that is inversely proportional to the effective number of samples in order to capture the diminishing marginal benefits of incrementing the samples of a class [50]. The key idea behind effectiveness is incrementing the data sample with unique instances rather than near-duplicates. Due to the intrinsic similarities among real-world data, increasing the sample size of a class/label might not necessarily improve model-performance. Effective samples E_n is computed by (24).

$$E_n = \frac{1-\beta}{1-\beta^{n_y}}, \beta = \frac{N-1}{N} \tag{24}$$

N is dataset size and n_y is the sample size of label y, $\beta^{n_y} = \frac{n_y - 1}{n_y}$. CB loss is therefore computed as shown in (25).

$$\text{CB} = E_n \cdot \text{ODP}_{\text{loss}} \tag{25}$$

Cost sensitive weighting loss functions to mitigate the imbalance distribution of the outcome types or labels. All functions apply a scaling factor to re-weight each sample during learning.

FOCAL LOSS (FL): FL assigns higher weights to harder examples and lower ones to the easier examples [130]. It introduces a scaling factor $(1-p)^{\lambda}$. λ is a focusing parameter in the loss function which decays to zero as the confidence in the correct class increases hence automatically down weighting the contribution of easy examples in the training and rapidly focusing on harder examples. FL is computed by (26).

$$\text{FL} = -\alpha_y (1 - P_y)^{\lambda} \cdot \text{ODP}_{\text{loss}} \tag{26}$$

where α is a weighting factor, $\alpha \in [0, 1]$, α_y is set to $\frac{1}{N_y}$, N_y is the number of training samples for label y, P_y is the probability of ground truth label y. I do not hypertune the focusing parameter λ, and instead set it to $\lambda = 2$ based on having achieved good results in examples presented in [50].

Using train and test splits whose statistics are provided in Table 14, the BiLSTM is trained with the different loss functions presented above on the OSD task and report the evaluation results on the test set in Table 15. The results show that both IIL variants and CB are quite competitive, however I chose IIL_2 particularly because it slightly outperforms all the other evaluated loss functions.

	EBM-COMET	EBM-NLP$_{rev}$
# of sentences	5193	40092
# of train/dev/test sentences	3895 / 779 / 519	30069 / 6014 / 4009
# of outcome labels	5	6
# of sentences with outcome spans in train/dev/test	1569 / 451 / 221	12481 / 4116 / 3257
Avg # of tokens per train/dev/test sentence	20.6 / 21.5 / 21.2	25.5 / 26.4 / 25.6
Avg # of outcome spans per sentence in train/dev/test	0.69 / 0.78 / 0.71	0.44 / 0.38 / 0.45

Table 14: Statistics summary of experimental datasets splits. Figures pertaining to Train, Dev and Test sets are separated by a forward slash accordingly.

4.3.2.3 Introducing an under-sampling hyper-parameter (US)

In addition to the cost-sensitive weighting, the majority class is undersampled by a particular percentage which is set a tunable hyper-parameter.

In addition to probing the loss function, I randomly under-sample the majority class of the dataset by a specified percentage that is later tuned as a hyper-parameter. At this stage, the objective of the ODP-tagger is to minimize the Imputed Inverse loss, particularly the smoothed version (IIL_2 (27)) because its outstanding results as discussed in the preceding section,

$$IIL_2 = -\frac{1}{\sqrt{N_y}} \sum_{(S,L)\in\mathcal{T}} \sum_{i}^{n} p(y_i|\boldsymbol{w}_i) \tag{27}$$

To finalize the expedition of determining which components to use in the ODP-tagger, I perform experiments in which I incrementally

	EBM-NLP$_{rev}$
BiLSTM*	27.0
BiLSTM + IIL_1	37.0
BiLSTM + IIL_2	38.0
BiLSTM + CB	37.0
BiLSTM + FL	19.0

Table 15: F1 % scores in the OSD task for various cost-sensitive loss functions on the EBM-NLP$_{rev}$ corpus. BiLSTM* implies the model was training with default ODP$_{loss}$ objective as shown in (22)

add the best performing components to ODP-tagger and monitor its performance in the OSD task. Table 16 results are emblematic of the positive impact each of the different components had in architecting the ODP-tagger. I observe cumulative gains in performance of 5.4%, 3.2% and 2.1% upon adding POS_{St}, $W2V_{Pb}$ and IIL_2 respectively. On the otherhand, adopting US_{50} and replacement of the softmax with a CRF for classification lead to slight improvements of 0.4% each.

I am aware that the improvements narrated above can dramatically change given new splits of the data, particularly the slight improvements brought about by US_{50} and the CRF. Therefore, to account for this, I check for the robustness of the improvements brought about by US_{50} and the CRF by measuring performance across 5 different randomly split train and test sets. The mean and (standard deviation) across the 5 experiments of the random splits are reported in Exps 7, 8 and 9. Results obtained in 8 and 9 show that both US_{50} and the CRF respectively lead to substantial improvements in performance when added to the ODP-tagger. Later on, multiple hyperparameters are tuned to obtain the optimal parameter settings (Table 17) for fine-tuning and feature extraction experiments.

4.3.3 Training

Components that ODP-tagger is augmented with i.e. POS, W2V embeddings, Imputed Inverse loss and Undersampling all lead to substantial gains in OSD achieved using ODP-tagger

Both sets of models (fine-tuned and feature-based) are evaluated on the two datasets, EBM-COMET and the EBM-NLP_{rev}. These datasets

Exps	Model	EBM-NLP_{rev}
1	BiLSTM*	32.5
2	BiLSTM* + POS_{St}	37.9
3	BiLSTM* + POS_{St} + $W2V_{Pb}$	41.1
4	BiLSTM + POS_{St} + $W2V_{Pb}$ + IIL_2	43.2
5	BiLSTM + POS_{St} + $W2V_{Pb}$ + IIL_2 + US_{50}	43.6
6	BiLSTM + POS_{St} + $W2V_{Pb}$ + IIL_2 + US_{50} + CRF	44.0
7	BiLSTM + POS_{St} + $W2V_{Pb}$ + IIL_2	42.8 (1.5)
8	BiLSTM + POS_{St} + $W2V_{Pb}$ + IIL_2 + US_{50}	43.2 (1.9)
9	BiLSTM + POS_{St} + $W2V_{Pb}$ + IIL_2 + US_{50} + CRF	44.3 (1.4)

Table 16: F1 % scores in the OSD task resulting from incrementally augmenting a BiLSTM with various components to build the ODP-tagger. BiLSTM* implies the model was training with default ODP_{loss} objective as shown in (22), POS_{St} denotes POS tagging by Stanford CoreNLP tagger, $W2V_{Pb}$ denotes Word2Vec trained using PubMed articles (Only non-contextual embeddings are tested in this investigation because they have smaller dimensions), IIL_2 denotes Imputed Inverse loss, US_{50} denotes Undersampling majority class by 50%. Exps 1-5 use a softmax classifier which is replaced by a CRF in 6. Exps 7-9 report the mean and (standard deviation) over 5 random train/test splits.

are each partitioned as follows, 75% for training (train), 15% for development (dev) and 10% for testing (test) as shown in Table 14. I exploit the large size of EBM-NLP$_{\textbf{rev}}$ and use its dev set to tune hyperparameters for the ODP-tagger and fine-tuned models (Parameter settings in Table 17). Each model is trained on a train split of a particular dataset and evaluated on the corresponding test split culminating into results shown in Table 18. I use a simple powerful NLP python framework called FLAIR[4] to extract word embeddings from all the BERT and FLAIR variants, and AllenAI[5] for BioELMO. Dimensions of the extracted BioFLAIR and BioELMO embeddings are very large, i.e. 7672 and 3072 respectively, which would most likely overwhelm my memory and power-constrained devices during training. Therefore, I apply Principal component Analysis (PCA) dimensionality reduction technique to reduce their dimensions to half their original sizes while preserving semantic information [176]. Alongside these embeddings, I evaluate non-contextual embeddings which are obtained by training word2vec (W2V) embedding algorithm [149] on 5.5B tokens of PubMed and PMC abstracts. Python and PyTorch[6] deep learning framework are used for implementation, which together with the datasets are made publicly available.[7]

Fine-tuning		
	Tuned range	Optimal
Learning rate	[1e-5,1e-4, 1e-3, 1e-2]	1e-5
Train Batch size	[16, 32]	32
Epochs	[3, 5, 10]	10
Sampling % (US)	[50, 75, 100]	100
Optimizer	[Adam, SGD]	Adam
ODP-tagger		
Learning rate	[1e-4, 1e-3, 1e-2, 1e-1]	1e-1
Train Batch size	[50, 150, 250, 300]	300
Epochs	[60, 80, 120, 150]	60
Sampling % (US)	[10, 25, 50, 75]	50
Optimizer	[Adam, SGD]	SGD

Table 17: Hyper-parameter tuning details in the feature extraction approach for the fine-tuned CLMs and the ODP-tagger (feature extraction).

4.4 EVALUATION EXPERIMENTS AND RESULTS

Results shown in Table 18 firstly reveal the superiority of fine-tuning based adaptation in comparison to the feature extraction adaptation

4 https://github.com/flairNLP/flair
5 https://github.com/allenai/bilm-tf
6 https://pytorch.org/
7 https://github.com/MichealAbaho/ODP-tagger

that uses the ODP-tagger. The best performance across both set-ups is obtained when BioBERT is fine-tuned on the EBM-COMET dataset. However, SciBERT is observed to outperform BioBERT in the ODP-tagger set-up on the EBM-COMET dataset. Secondly, the non-contextual W2V embeddings produce competitive performance on EBM-NLP$_{\textbf{rev}}$, however, they perform significantly lower than the CRS on EBM-COMET. BioFLAIR and ClinicalBERT were the least performing models. For BioFLAIR, I hypothesize that, (1) pre-training on a relatively smaller corpus, (2) it being of much less depth (1-layered BiLSTM) compared to multi-layered BERT and ELMo and (3) downsizing its embeddings using PCA dimensionality reduction are reasons that led to its low performance. For ClinicalBERT, I attributed its struggles to the nature of the corpora on which it is pre-trained. Unlike BioBERT, SciBERT and BioELMo which are pre-trained on PubMed text which mostly contains clinical trial abstracts that more often report health outcomes, ClinicalBERT is pre-trained on clinical notes associated with patient hospital admissions [99]. Overall, CRS provided by BioBERT were the best performing embeddings, consistently outperforming all the other CRS across both setups and both datasets. An additional insight drawn was that, performance on the EBM-NLP$_{\textbf{rev}}$ dataset is lower compared to that achieved on EBM-COMET. This was attributed to the annotation inconsistencies in the original EBM-NLP, some of which were resolved and documented in [1]. Another aspect I closely monitored was the runtime. Using a TITAN RTX 24GB GPU, the average runtime for the fine-tuning experiments on EBM-COMET and EBM-NLP$_{\textbf{rev}}$ was 7 and 12 hrs respectively. On the other-hand, feature extraction (ODP-tagger) experiments were much longer consuming 20 and 36 hours respectively on the same datasets. Overall, the results suggest and recommend fine-tuning as a preferred approach for adapting CRS to the OSD task. Furthermore, given their dominant per-

NLP libraries FLAIR and AllenNLP are used to extract features from BERT, FLAIR variants and BioELMo respectively. These features are then either fine-tuned or trained using ODP-tagger.

Fine-tuning			Feature extraction		
Model	EBM-NLP$_{\textbf{rev}}$	EBM-COMET	Model	EBM-NLP$_{\textbf{rev}}$	EBM-COMET
W2V	_	_	ODP-tagger + W2V	44.0	59.3
BERT	51.8	75.5	+BERT	43.2	64.2
ELMO	49.6	71.4	+ELMO	43.0	61.2
BioBERT	**53.1**	**81.5**	+BioBERT	**48.5**	69.3
BioELMO	52.0	75.0	+BioELMO	46.5	62.9
BioFLAIR	51.4	76.7	+BioFLAIR	40.7	60.5
SciBERT	52.8	77.6	+SciBERT	48.1	**70.4**
ClinicalBERT	51.0	68.5	+ClinicalBERT	45.2	65.7
Bio+ClinicalBERT	51.0	68.3	+Bio+ClinicalBERT	45.8	66.3
Bio+Disc Summary BERT	51.0	70.0	+Bio+Disc Summary BERT	46.1	68.4

Table 18: Macro-average F1 scores obtained from generic CLMs and their respective In-domain (biomedical) versions for both fine-tuning and feature extraction (ODP-tagger) for token-level detection of outcome spans from both datasets.

formance detailed in the preceding paragraph, this work nominates BioBERT and SciBERT as ideal CRS for the OSD task.

4.4.1 Full outcome span detection

Full outcome span detection is extended analysis I perform to evaluate how well the models detect full mentions of outcome spans.

Motivated by Aken et al. [8] who indicate that accurate fine-grained information is beneficial in the medical domain, I examine the extent to which the models detect precise mentions of full outcome spans. To achieve this, I investigate how well the best performing models (Fine-tuned+BioBERT+EBM-COMET and Fine-tuned+BioBERT +EBM-NLP$_{\textbf{rev}}$ from Table 18) can detect full mentions of outcome spans or otherwise exact matches of outcome spans in prediction results. I use a strict criteria to evaluate full mention of outcome spans, where a classification error FN (False Negative) accounts for the number of full outcome spans the model fails to detect, which includes partially correctly detected spans i.e. some of their tokens were misclassified. In addition to Precision, Recall (also known as Sensitivity) and F1 measure, I report Specificity (True Negative Rate (TNR)) for extended analysis of the model performance. In Table 19, it is noticeable that F1 of the best models drops from 53.1 to 52.4 for EBM-NLP$_{\textbf{rev}}$ and 81.5 to 69.6 for EBM-COMET. These results signal the difficulty the models have in identifying full outcome spans, especially with the EBM-NLP$_{\textbf{rev}}$ dataset. Specificity on the other hand is very high for both datasets simply because it is calculated as a TNR, in which case True Negatives (non-outcomes) are certainly so many because they are precisely individual words and therefore are counted word by word as opposed to True positives (actual outcome spans) that can consist of multiple words.

4.4.2 Error Analysis

I further investigate the errors from the best performing models, Fine-tuned+BioBERT+EBM-COMET and ODP-tagger+SciBERT+EBM-COMET

		P	R	S	F
EBM-NLP$_{\textbf{rev}}$		55.1	51.2	99.6	53.1
EBM-NLP$_{\textbf{rev}}$	Full outcome spans	53.7	51.2	99.2	52.4
EBM-COMET		76.1	87.7	99.4	81.5
EBM-COMET	Full outcome spans	60.8	81.3	98.0	69.6

Table 19: Results of Precision (P), Recall/Sensitivity (R), Specificity (S) and F1 of evaluating best performing fine-tuned models (Fine-tuned+BioBERT+EBM-NLP$_{\textbf{rev}}$ and Fine-tuned+BioBERT+EBM-COMET) in OSD for precise mentions of full outcome spans. The non bold-faced row are results originally obtained without full outcome span evalution.

in the fine-tuning and feature extraction setups respectively. Table 20 shows examples of outputs of both models for the OSD task given an input sentence with known actual outcome spans underlined. Spans are coloured blue to indicate correctly identified whereas red indicates the opposite. As seen in the table, the fine-tuned model correctly detects all full outcome spans in the first example sentence i.e. Precision (P), Recall/Sensitivity (R) are 100%, whereas tagger only detects 3/4 outcomes, hence P is 100%, R is 75%. Neither of the models correctly capture full mention of the outcome spans in the second example, they incorrectly predict "duration of" to not belong to the outcome span. While traditionally, sequence labelling and NER results would be a P of 100% and R of 50% for correct prediction of 2/4 tokens for both fine-tuned and ODP-tagger model, in the strict full outcome evaluation, P and R are 0%, because some tokens in the full outcome span are mis-classified in both models i.e. True positives = 0. Similarly, in the third example, the fine-tuned model achieves P of 100% and R of 60% for correct prediction of 3/5 tokens in the tradi-

Method		Abstract sentence	Full outcome span	
	Input sentence	*Among patients who received sorafenib, the most frequently reported **adverse events** were grade 1 or 2 events of rash (73%), **fatigue** (67%), **hypertension** (55%) and **diarrhea** (51%).*	***- adverse events*** ***- rash***	***- fatigue*** ***- hypertension*** ***- diarrhea***
BioBERT+ EBM-COMET	Output	*Among patients who received sorafenib, the most frequently reported **adverse events** were grade 1 or 2 events of **rash** (73%), **fatigue** (67%), **hypertension** (55%) and **diarrhea** (51%).*	***- adverse events*** ***- rash***	***- fatigue*** ***- hypertension*** ***- diarrhea***
ODP-tagger+ SciBERT +EMB-COMET	Output	*Among patients who received sorafenib, the most frequently reported adverse events were grade 1 or 2 events of **rash** (73%), **fatigue** (67%), **hypertension** (55%) and **diarrhea** (51%)..*	***- fatigue*** ***- diarrhea***	***- hypertension***
	Input sentence	*The average **duration of operating procedure** was 1 hour and 35 minutes.*	***- duration of operating procedure***	
BioBERT+ EBM-COMET	Output	*The average **duration of operating procedure** was 1 hour and 35 minutes.*		
ODP-tagger+ SciBERT +EMB-COMET	Output	*The average **duration of operating procedure** was 1 hour and 35 minutes.*		
	Input sentence	*The objective of this study was to evaluate **right heart size** and **function** assessed by echocardiography during long term treatment with 16.5cmriociguat.*	***- right heart size*** ***- right heart function***	
BioBERT+ EBM-COMET	Output	*The objective of this study was to evaluate **right heart size and function** assessed by echocardiography during long term treatment with riociguat.*	***- right heart size***	
ODPtagger+ SciBERT+ EMB-COMET	Output	*The objective of this study was to evaluate **right heart size and function** assessed by echocardiography during long term treatment with riociguat.*		

Table 20: Example outcome detection outputs from best fine-tuned BioBERT and ODP-tagger+SciBERT models.

tional NER evaluation, whereas for the strict full outcome span evaluation, R is 50% because only 1/2 full outcome spans are detected.

I attribute these errors to the length of some outcome spans with some containing extremely common words such as prepositions (*"of"*). Additionally, I note that the contiguous outcome span annotations (containing several outcomes sharing terms e.g. "right heart size and function" in the third example) are rare and therefore will be harder instances for the model to correctly classify.

4.4.3 Evaluation on the original EBM-NLP

	P	I	O
Logreg	45.0	25.0	38.0
Lstm-crf	40.0	50.0	48.0
Brockmeier et.al [26]	70.0	56.0	70.0
Fine-tuned BioBERT	**71.6**	**69.0**	**73.1**
Fine-tuned BioBERT – Full outcome span mentions	61.6	64.0	53.1

Table 21: F1 scores of token level detection of PIO elements reported for EBM-NLP hierarchical labels dataset by the EBM-NLP [161] leader board.

I additionally fine-tune the best model (Fine-tuned+BioBERT+EBM-NLP$_{\textbf{rev}}$) for the task of detection of all PIO elements (Participants (P), Interventions (I) and Outcomes (O)) in the original EBM-NLP (hierarchical version) dataset. To be consistent with the original EBM-NLP paper [161], I consider the token-level detection of the PIO elements task in their work, comparing their evaluation results for hierarchical labels with those I obtain by fine-tuning the best model. Using their published training (4670) and test (190) sets of the starting spans, I observed the model outperforms the published leader board results[8] and the recent results published by Brockmeier et al [26] (Table 21). This improvement is attributable to the fact, unlike the LSTM-CRF and Logreg models in previous SOTA scores, BioBERT's has an internal capability to encode information using self-attention mechanisms and the MLM to generate context-sensitive representations of words.

The fine-tuned BioBERT outperforms leader board results on the original EBM-NLP at the point of performing the experiments.

4.4.4 Outcome span length

To further understand my results, I investigated how well the best model Fine-tuned+BioBERT+EBM-COMET and ODP-tagger+SciBERT +EBM-COMET (Feature-extraction) detected outcome spans of vary-

8 https://ebm-nlp.herokuapp.com/

ing lengths. This investigation is conducted because health outcome spans range from a single word to multiple words. I calculate a prediction accuracy as number of correctly predicted outcome spans of length x/number of all outcome spans of length x, where x ranged from 1-10. As observed in 7, the fine-tuned model slightly outperforms the ODP-tagger especially for outcome spans having 3-6 words (i.e. 3-6 entity span length). However, it is also clear that both models struggled to accurately detect outcome spans containing 7 or more words.

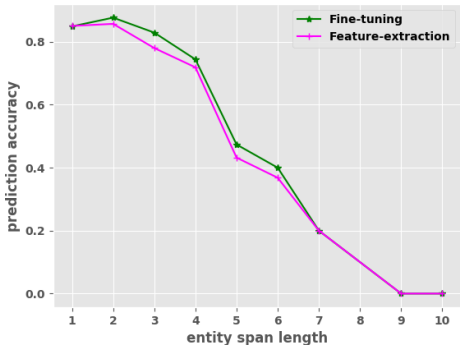


Figure 7: Prediction accuracy per entity text-span length.

4.5 DISCUSSION AND SUMMARY

In this chapter, extensive analysis has been performed on how CRS provided by various pre-trained biomedical CLMs perform in detecting outcomes (OSD) from biomedical text in two datasets annotated with outcomes, EBM-NLP$_{\mathbf{rev}}$) and EBM-COMET. The main aim in this chapter was to not only inspect which CRS are suitable for OSD, but additionally inspect which paradigm between fine-tuning and feature extraction (feature based) adaptations would lead to the best transferability of these CRS for the OSD task.

To this end, the chapter designed an adaptation framework in which I fine-tuned pre-trained CRS (BioBERT, SciBERT, ClinicalBERT, BioELMo and BioFLAIR) and additionally transferred and trained these CRS in a customly built neural model (ODP-tagger) for the OSD task. In the fine-tuning adaptation approach, I use a task-specific classification layer on top of each of the different architectures providing the CRS. On the other hand, for the feature extraction adaptation approach, I systematically build a neural model by augmenting a BiLSTM with clinically oriented POS features, a cost sensitive loss function, undersampling component and a CRF classification layer. The main conclusion drawn was that, fine-tuning adaptation is the ideal setting for transferring pre-trained biomedical features for the OSD task. Fine-tuned models consistently outperform and converge faster than the corresponding feature extraction models. Because of their stronger

performance in comparison to the other CRS across both adaptation setups and datasets, a consensus is arrived at that pre-trained BioBERT and SciBERT CRS are best suited for detecting outcomes in biomedical text. In the future, further insights would be obtained by comparing freezing the CLM weights in the fine tuning architecture against standard fine tuning, as well as fine tuning CLM weights in the ODP-Tagger architecture against standard feature extraction.

Moreover I extended the analysis to investigate how well the best performing models detect precise mentions of full outcome spans i.e. an inspection of how efficient the model is in detecting exact matches of outcome spans in the text. In this analysis, I discover that the performance of the model deteriorates by about 1.3% on the EBM-NLP$_{\textbf{rev}}$ dataset and 11.9% on the EBM-COMET from the original evaluation scores that did not target exact matches or full outcome span mentions in the predicted outcomes. This performance decline is attributed to the strict evaluation criteria which absolutely never rewarded the models for any partial correct predictions. This insight informed the evaluations I perform in the future chapters because of accurate detection of full mention of granular outcome spans is beneficial for clinicians searching for this information [8].

To validate the best performing model, I compared its performance in PIO extraction (detecting all PIO elements) in the original EBM-NLP to the leaderboard results on EBM-NLP[9] (hierarchical labels) and SOTA published by Brockmeier et al. [26]. I report gains of 1.6% for P, 10.0% for I and 3.1% for O over recent SOTA F1 scores.

Further analysis including error analysis and an inspection of how the length of outcome spans varies with the performance in the OSD task. Several errors are observed in detecting long health outcome spans which is further proven in an illustration that shows that accuracy in detecting outcome spans is inversely proportional to the length of the outcome span.

Chapter 3 embarked on refining outcome annotations and evaluating them in an OC task. This chapter has illuminated the OSD task proving that fine-tuning is a preferrable approach to transferring pre-trained features. In the next chapter, I attempt to learn from both token- and sentence-level information to jointly achieve the two tasks.

9 https://ebm-nlp.herokuapp.com/

5

JOINT SPAN DETECTION AND CLASSIFICATION FOR HEALTH OUTCOMES

5.1 INTRODUCTION AND BACKGROUND

In earlier chapters and most especially the previous Chapter 4, I have elaborated how prior work on health outcome detection (HOD) modelled it as either an *Outcome Span Detection* (OSD) task or an *Outcome classification* (OC) task. The goal in OSD is to detect a continuous span of tokens that indicate a health outcome [26, 161], where as the goal in OC is to classify a text span into a pre-defined set of outcome types/classes depending on an outcome it mentions. The two tasks are, however, highly correlated i.e. local token-level information relevant in OSD enables us to make accurate global sentence-level outcome predictions relevant in OC, and vice versa.

Decoupling of OSD and OC would imply that, the mutual compatibility constraints between outcome spans and their types will be lost

Given the ideal result of the two tasks i.e. an outcome in form of an entity text span for OSD and a pre-defined outcome type for OC, the task can be formulated as a downstream sequence labelling NER task, where entities are annotated with BIO labels of multiple entity types [199]. Whereas this conventional sequence labelling approach would work fine, it has limitations in a multi-label scenario where an entity can be associated with multiple labels. The number of B, I and or O labels for a single token within an entity span can easily exponentially increase with an increasing number of entity-type labels for that entity span. More so, the risk of model forgetfulness for low occurrence entity-type labels is increased [139]. To avoid these scenarios, given the multi-labeled outcome span annotations in the datasets investigated in this chapter, I cast the overall outcome detection task as a joint learning task that optimizes for NER and Text classification. Nevertheless, I later formulate the task in the conventional sequence labelling approach for NER tasks in order to further realise any performance gains or losses in the OSD task.

Besides, an outcome type predicted for a text span in a sentence must be consistent with the other outcome spans detected from the same sentence, while the outcome spans detected from a sentence must also be compatible with their outcome types. From a modelling perspective, decoupling of OSD and OC would imply that, the mutual compatibility constraints between outcome spans and their types will be lost, hence limiting the potential performance gains attributable to this compatibility in both tasks. Furthermore, this decoupled approach exposes the outcome detection process to task error propagation i.e. the error/s made in the OSD task will be propagated to the OC

task and vice versa if the two tasks are conducted in tandem within a pipelined framework such as that adopted by [33].

Joint learning models have been proposed to alleviate the aforementioned disadvantages for various related domain-specific tasks such entity extraction and relation classification [17, 36, 104], slot filling and topic classification [143], aspect extraction (AE) and aspect sentiment classification (ASC [103, 230] etc. Motivated by the recent success in joint learning strategies mentioned in the previous sentence as well as MTL models whose optimised parameters are shared across multiple tasks [147, 174], this chapter proposes a method that exploits the global structural correspondences between word- and sentence-level information to *simultaneously* perform OSD and OC. In addition to injecting contextual information to hidden vectors, I use label attention to appropriately weight both word and sentence level information.

I propose a method that exploits the global structural correspondences between word- and sentence-level information to simultaneously *perform OSD and OC.*

The rest of this chapter begins by discussing the problem setup, task formulation and a preamble of the proposed joint OSD and OC method in Section 5.2. Following this section, I propose the LCAM to simultaneously learn label-attention weighted representations at word- and sentence-level in Section 5.3. These representations are then evaluated in Section 5.4. Ablation experiments and investigations on these representations are additionally included in Section 5.4. The chapter concludes with a summary of the highlights and achievements of the proposed method in Section 5.6.

5.2 JOINT OSD AND OC CHALLENGE

In this section, I frame HOD as a joint task that involves simultaneously performing OSD and OC. Ideally, a strong joint OSD and OC system should be able to effectively achieve both tasks without compromising the performance in standalone settings in which the two tasks are separately performed.

Health Outcome Detection (HOD) Task formulation

Given a sentence $s = w_1, \ldots, w_M$ extracted from a clinical trial abstract, the goal of HOD is to develop a joint learning model that identifies an outcome span $o_d = b_i, \ldots, b_N$ (i.e OSD), and subsequently predicts a plausible outcome type $t(o_d) \in \mathcal{Y}$ for o_d (i.e. OC), where $1 \leqslant i \leqslant N \leqslant M$, and $\mathcal{Y}$ is a predefined set of outcome types.

To illustrate the distinction between the OSD, OC and Joint OSD & OC tasks, I present two examples shown in Table 22. Specifically, in the first sentence, OSD extracts all outcomes i.e. *wheezing* and *shortness of breath*, OC classifies the text into an outcome type, Physiological, and then Joint OSD & OC extracts an outcome span and classifies it concur-

rently i.e. it extracts *wheezing* and also classifies it as a Physiological outcome.

sentence	There were no significance between-group differences in the incidence of **wheezing** or **shortness of breath**
OSD	Outcomes: **wheezing**, **shortness of Breath**
OC	Outcome type: Physiological
Joint OSD & OC	Outcomes-Outcome type **wheezing**-Physiological **Shortness of Breath**-Physiological
sentence	Cumulative incidence and relative risks with 95% confidence intervals for **death from any cause**, **death from prostate cancer**, and metastasis were estimated in intention-to-treat and per-protocol analyses.
OSD	Outcomes: **death from any cause**, **death from prostate cancer**
OC	Outcome type: Mortality
Joint OSD & OC	Outcomes-Outcome type **death from any cause**-Mortality **death from prostate cancer**-Mortality

Table 22: Comparing the output of the three separate HOD tasks given two sample sentences. OSD retrieves the outcome spans, OC classifies the text span into a set of outcome types, and Joint OSD & OC retrieves outcomes and classifies them into outcome types.

5.2.1 Joint learning and evaluation approach.

LCAM is designed to jointly learn contextualised label attention-based distributions at word- and sentence-level in order to capture which label/s a word or a sentence is more semantically related to.

I propose LCAM, a sequence-to-sequence-to-set (SEQ2SEQ2SET) model extensively discussed in Section 5.3, which uses a single encoder to represent an input sentence and two decoders, one for predicting the label for each word in OSD and another for predicting the outcome type in OC. LCAM is designed to jointly learn contextualised label attention-based distributions at word- and sentence-levels in order to capture which label/s a word or a sentence is more semantically related to. I call them contextualised because they are enriched by global CRS of the abstracts to which the sentences belongs. Label attention incorporates label sparsity information and hence semantic correlation between documents and labels.

A baseline BiLSTM and or clinically informed $\text{BERT}_{\text{base}}$ [54] models are used at the encoding stage of the model and sigmoid prediction layers are used at the decoding stage. I also use a Multi-label Prediction (MLP) layer for the two tasks (i.e. OSD and OC), with a relaxed constraint at token-level that ensures only the top (most relevant) prediction is retained, whereas all predicted (relevant) outcome types are retained at the sentence-level during OC. I use a MLP layer because

some annotated outcomes belong to multiple outcome types. For example, *depression* belongs to both *"Physiological"* and *"Life-Impact"* outcome types.

The models are evaluated on the tasks by reporting the macro-averaged F1. Inaddition to the macro-F1, I visualise ranking metrics pertaining to MLP, in order to compare my model to related work for MLP. The metrics of focus include precision at top n P@n (fraction of the top n predictions that is present in the ground truth) and Normalized Discounted Cumulated Gain at top n (nDCG@n).

5.2.2 Data

Chapter 3 introduced EBM-NLP$_{\textbf{rev}}$ (a revised version of EBM-NLP [161]) and EBM-COMET, whose annotation was premised on classifications in the standard outcome classification taxonomy proposed by Dodd et al. [55]. The chapter also proposed an unsupervised label alignment method to denoise outcome classification label annotations in EBM-NLP$_{\textbf{rev}}$ by identifying and aligning parallel annotations across the EBM-NLP and EBM-COMET. After the label denoising, both datasets had a reconciled classification label set i.e. the set of outcome type labels in EBM-NLP$_{\textbf{rev}}$ was exactly the same as that in EBM-COMET.

In addition to these two datasets, I merge them to create a large dataset, EBM-COMET+EBM-NLP$_{\textbf{rev}}$ to use in evaluating my joint learning approach. The merger of the two datasets is also for the purpose of evaluating the label alignment approach earlier introduced in Section 3.4 because I hypothesise that the merged dataset would improve performance obtained on the original independent datasets. All three datasets are used during evaluation, with each one being randomly split into two, where 80% is retained for training and 20% for testing as shown in Table 23.

	EBM-COMET	EBM-NLP	EBM-COMET + EBM-NLP$_{\textbf{rev}}$
# of Abstracts	300	5000	5300
# of sentences	5193	40092	45285
# of outcome labels	5	6	5
avg sentence length	21.0	26.0	25.0
# of Training sentences	4155	32074	36229
# of Testing sentences	1038	8018	9056

Table 23: Datasets statistics rounded off to zero decimal

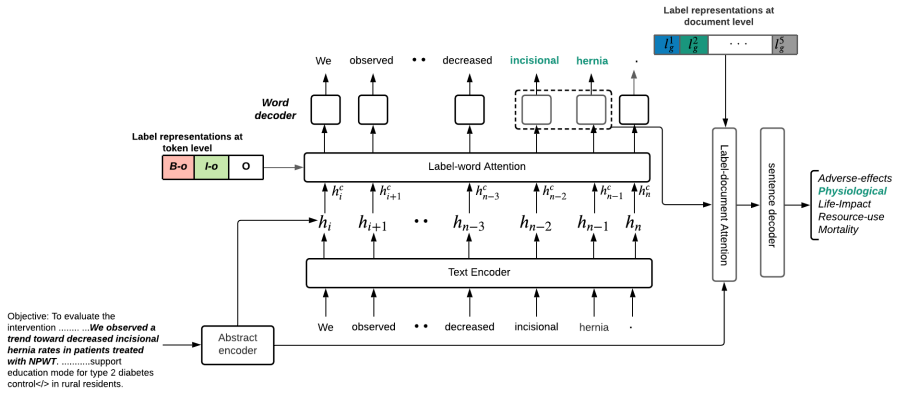


Figure 8: Illustration of the LCAM Architecture. It encodes a sequence of tokens of a sentence within an abstract, generates contextualised representations by adding a global representation of the abstract at word- and sentence-level. Two attention layers are used to aid generation of label-aware representations used to decode labels at word-level for OSD and sentence-level for OC.

5.3 LABEL CONTEXT-AWARE ATTENTION MODEL (LCAM)

LCAM is a sequence-to-sequence-to-set architecture. An output sequence is generated for a given input sequence, and then a set of non-sequential items is generated for the output sequence.

Figure 8 illustrates an end-to-end SEQ2SEQ2SET architecture of the LCAM model. It depicts a two-phased process to achieve classification at token and sentence level. In phase 1, input tokens are encoded into representations which are sent to a decoder which is a sigmoid layer to predict a label for each word, hence OSD. Subsequently, in phase 2, the token-level representations are used to generate individual outcome span representations, which are sent to another decoder (sigmoid layer) that is used to predict the label/s for each outcome span, hence OC. I use MLP for the OC task because some outcomes are annotated with multiple outcome types.

5.3.1 Outcome Span Detection (OSD)

Given a set of sentences $\mathcal{S} = \{s_i\}_{i=1}^{|\mathcal{S}|}$ within an abstract a, each s_i having N words, $s_i = w_1, \ldots, w_N$, with each word tagged to a label l_w using the BIO tagging scheme [182]. OSD aims to extract one or more outcome spans within s_i. For example, in Figure 8, OSD extracts the outcome span *"incisional hernia"* given the input sentence.

ENCODER: In the OSD task setting, I initially implement a baseline LCAM using a BiLSTM to encode input tokens (that are represented by d-dimensional word embeddings obtained using GloVe [166][1]) into hidden representations for every word within an input sentence. I then consider generating each input word's hidden representation us-

1 https://github.com/stanfordnlp/GloVe

Input token embeddings are encoded by a BiLSTM used as as baseline and later on BioBERT in a seperate model. An abstract representation is then added to each encoded hidden state to make it context-aware.

ing a pre-trained clinically informed $\text{BERT}_{\text{base}}$ model called BioBERT [121]. The LCAM model learns (28),

$$\begin{aligned} \mathbf{h}_n &= \text{BiLSTM}(w_n), \\ \mathbf{h}_n &= \text{BioBERT}(w_n) \end{aligned} \tag{28}$$

where $w_n \in s_i$, $\mathbf{h}_n \in \mathbb{R}^{k\times 1}$ and k is the dimensionality of the hidden state. The upper equation under 28 is used for a BiLSTM Text encoder and the lower for a BioBERT Text encoder.

ABSTRACT HIDDEN STATE CONTEXT: To make the hidden state representation context-aware, I add a compound representation of the abstract in which the sentence containing w_n belongs using (29).

$$\mathbf{h}^c_n = \mathbf{h}_n + f(\text{AbsEncoder}(a)) \tag{29}$$

where f is a function computing the average pooled representation of the encoded abstract, $\text{AbsEncoder} \in \{\text{BiLSTM}, \text{BioBERT}\}$, $\text{AbsEncoder}(a) \in \mathbb{R}^{k\times|a|}$, $|a|$ is the length of the abstract (measured by the number of tokens contained in it) and $f(\text{AbsEncoder}(a)) \in \mathbb{R}^{k\times 1}$.

5.3.2 Label-word attention

Two different attention scores are computed, the first is to enable the model pay appropriate attention to each word when generating the overall outcome span representation. Then the second attention score, is to allow the words interact with the labels in order to capture the semantic relation between them, hence making the representations more label-aware. To obtain the first attention vector $\mathbf{A}^{(1)}$, I use a self-attention mechanism [9, 131] that uses two weight parameters and a hyper parameter b that can be set arbitrary,

Two attention vectors are computed, a self attention vector to capture word-to-word interaction a label attention vector to capture label-word interaction.

$$\mathbf{A}^{(1)}_n = \text{softmax}(\mathbf{W}\tanh(\mathbf{V}\mathbf{h}^c_n)) \tag{30}$$

where $\mathbf{W} \in \mathbb{R}^{|l_w|\times b}$, $\mathbf{V} \in \mathbb{R}^{b\times k}$ and $\mathbf{A}^{(1)} \in \mathbb{R}^{|l_w|\times 1}$. $|l_w|$ is the number of token-level labels. Furthermore, I obtain a label-word attention vector $\mathbf{A}^{(2)}$ using a trainable matrix $\mathbf{U} \in \mathbb{R}^{|l_w|\times k}$. Similar to the interaction function Du et al. [59] use, this attention is computed in (31) as the dot product between the $\mathbf{h}^c_n$ and $\mathbf{U}$,

$$\mathbf{A}^{(2)}_n = \mathbf{U}\mathbf{h}^c_n \tag{31}$$

where $\mathbf{A}^{(2)}_n \in \mathbb{R}^{|l_w|\times 1}$.

LABEL-WORD REPRESENTATION: The overall representation used by the decoder for classification of each token is obtained by merging the two attention distributions from the previous paragraphs as shown by (32),

$$\mathbf{E}^{t_l}_n = \mathbf{A}^{(1)}_n \mathbf{h}^{c^\top}_n + \mathbf{A}^{(2)}_n \mathbf{h}^{c^\top}_n \tag{32}$$

where $\mathbf{E}_n^{t_l} \in \mathbb{R}^{|l_w| \times k}$, denotes the token-level (t_l) representation.

The training objective is to maximise the probability of a singular ground truth label and minimise a cross-entropy loss,

$$L_{osd} = -\sum_{n=1}^{N}\sum_{i=1}^{|l_w|} y_{n,i}\log(\hat{y}_{n,i}). \tag{33}$$

where N is number of tokens in a sentence, $|l_w|$ is the number of labels.

5.3.3 Outcome Classification (OC)

OC predicts outcome types for the outcome spans extracted during OSD. Similar to what is done at token-level, I add an abstract representation (which is a mean pool of its token's representations) to add context to each tokens representation. An outcome span is represented by concatenating the vectors of its constituent words,

An outcome span representation is computed as a concatenation of all token-level representation for the constituent tokens added to an abstract representation.

$$\mathbf{O}_s = \bigoplus_{i=1}^{m}(\mathbf{E}_i^{t_l} + f(\text{AbsEncoder}(a))) \tag{34}$$

where m is the number of tokens contained in outcome span O_s. I adopt the aforementioned self-attention and label-word attention methods at sentence-level to aid extraction of an attention based sentence-level representation of an outcome as follows:

$$\mathbf{E}_s^{s_l} = \mathbf{A}^{(1)}\mathbf{O}_s + \mathbf{A}^{(2)}\mathbf{O}_s \tag{35}$$

where $[\mathbf{A}^{(1)}, \mathbf{A}^{(2)}] \in \mathbb{R}^{|l_s| \times m}$, $\mathbf{O}_s \in \mathbb{R}^{m \times k}$ and $s \geqslant 0$.

Given an outcome span representation $\mathbf{E}^{s_l}$, the training objective at sentence-level (s_l) is to maximize the probability of the set of terms,

$$\underset{\theta}{\text{argmax}} P(y = (l_s^1, l_s^2, ..., l_s^6) \in l_s | \mathbf{E}^{s_l}; \theta) \tag{36}$$

$$L_{oc} = -\sum_{i=1}^{|l_s|} y_i \log(\hat{y_i}) + (1 - y_i)\log(1 - \hat{y_i}) \tag{37}$$

where $y_i \in \{0, 1\}$, $\hat{y_i} \in [0, 1]$ $l_s \in$ {Physiological, Mortality, Life-Impact, Resource-use, Adverse-effects}. The overall joint model loss is:

$$L = L_{osd} + L_{oc} \tag{38}$$

5.3.4 LCAM Algorithm

To demonstrate the flow of the joint learning training, I use the pseudo code in algorithm 2 to show how I arrive at the joint model loss. For each token's hidden state (line 8), I compute a context aware hidden state by adding to it an encoded abstract representation (line 9) and then compute two attention scores (line 10 - 14) that both capture the contribution the token makes to each token-level label. These are then used together to generate a label-word representation (line 16) and all label-word representations forming a sentence (line 17) are used to compute an OSD loss using (33) (line 19). Once again I add context to the newly generated toke-level representations (line 20). For every outcome, I repeat steps in lines 10-14 to obtain label attention scores i.e. depicting the contribution the particular outcome phrase makes to each outcome-type label and these are used to obtain a label-document representation for the outcome (line 30). This representation is then used to compute the outcome classification loss (line 32). The loss I minimise in the joint learning is computed as shown by line 33.

5.4 EVALUATION EXPERIMENTS AND RESULTS

In this section, I present implementation details, experimental setups and the main results of evaluating the joint learning LCAM framework on the three datasets discussed in Section 5.2.2.

5.4.1 Implementation

For pre-processing the data, I first label each word in the sentences contained in an abstract with either one of {B, I, O}. Subsequently, to the end of each sentence, I include a list of outcome types corresponding to the outcome spans in the sentence. However, it is important to note that, not all sentences within an abstract had outcome spans. For example, the annotated sentence below contains outcome span "Incisional hernia" whose outcome type label (Physiological) is placed at the end of the sentence.

Each sample is annotated both at token level and at sentence level. The latter includes assigning outcome type labels for the constituent outcome spans.

*"We/**[O]** observed/**[O]** a/**[O]** trend/**[O]** toward/**[O]** decreased/**[O]** incisional/**[**__**B-outcome**__**]** hernia/**[**__**I-outcome**__**]** rates/**[O]** in/**[O]** patients/**[O]** treated/**[O]** with/**[O]** NPWT/**[O]** ./**[O]**". [[__**Physiological**__]]*

I tuned hyper-parameters using 20% of the training data of the merged dataset (EBM-NLP+EBM-COMET) as a development set. Table 24 shows the range of values (including the lower and upper bound) for which the LCAM-BioBERT (BioBERT used as the encoder) and Standalone models are tuned to obtain optimal configurations. The optimal settings are included under the optimal settings columns

Algorithm 2 LCAM Algorithm

Input: train data, **Output**: model weights
for abstract a in train data **do**
  Obtain $Abs = \text{AbsEncoder}(a)$
  for sent s in a **do**
    Obtain $\mathbf{H} = \text{Encoder}(s)$
    where $\mathbf{H} \in \mathbb{R}^{k \times n}$
    Initialise: an empty tensor S
    for $\mathbf{h}_n$ in $\mathbf{H}$ **do**
      $\mathbf{h}_n^c = \mathbf{h}_n + f(Abs)$
      Obtain $\boldsymbol{A}^{(1)} = \text{softmax}(\mathbf{W} \tanh \mathbf{V}\mathbf{h}_n^c)$
      where $\boldsymbol{V} \in \mathbb{R}^{b \times k}$, $\mathbf{W} \in \mathbb{R}^{|l_w| \times b}$,
      and $\boldsymbol{A}^{(1)} \in \mathbb{R}^{|l_w| \times 1}$
      Obtain $\boldsymbol{A}^{(2)} = \mathbf{U}\mathbf{h}_n^c$
      where $\mathbf{U} \in \mathbb{R}^{|l_w| \times k}$, $\boldsymbol{A} \in \mathbb{R}^{|l_w| \times 1}$
      label-word representation:
      $\mathbf{E}^{t_l} = \boldsymbol{A}^{(1)}\mathbf{h}_n^{c^\top} + \boldsymbol{A}^{(2)}\mathbf{h}_n^{c^\top}$
      $S = S \oplus \mathbf{E}^{t_l}$
    end for
    Compute Loss eqn 9 - L_{osd}
    $\forall \mathbf{E}^{t_l} \in S : \mathbf{E}^{t_l} = \mathbf{E}^{t_l} + f(Abs)$
    $\forall O_x \in S$, where $x \geqslant 0$ & $O_x \in \mathbb{R}^{m \times k}$
    i.e. outcome $\mathbf{O}_x$ has m tokens
    for outcome $\mathbf{O}$ in S **do**
      Obtain $A^{(1)} = \text{softmax}(\mathbf{W} \tanh(\boldsymbol{V}\mathbf{O}^\top))$
      where $\boldsymbol{V} \in \mathbb{R}^{b \times k}$, $\mathbf{W} \in \mathbb{R}^{|l_s| \times b}$
      and $\boldsymbol{A} \in \mathbb{R}^{|l_s| \times m}$
      Obtain $\boldsymbol{A}^{(2)} = \mathbf{U}\mathbf{O}^\top$
      where $\mathbf{U} \in \mathbb{R}^{|l_s| \times k}$, $\boldsymbol{A} \in \mathbb{R}^{|l_s| \times m}$
      label-document representation of an outcome:
      $\mathbf{E}^{s_l} = \boldsymbol{A}^{(1)}\mathbf{O} + \boldsymbol{A}^{(2)}\mathbf{O}$
    end for
    Compute Loss L_{oc} eqn 13
    minimise model loss $L = L_{osd} + L_{oc}$
  end for
end for

in the table. Experiments were performed using a Titan RTX 24GB GPU.

5.4.2 Setup

The Joint setup is concurrent sequence labelling (OSD) and sequence classification (OC) whereas the standalone setup, is OSD and OC performed separately. The former is achieved using (a) a Baseline model,

		Joint models	Standalone models
parameter	**Tuned range**	**Optimal**	**Optimal**
Train batch size	[8,16,32,64]	64	32
Eval batch size	[8,16,32,64]	16	8
Embedding dim			
- baseline	–	300	–
- BERT models	–	768	768
b	[150,200,250]	200	_
Optimizer	[Adam, SGD]	Adam	Adam
Epochs	[5,10,15]	10	10
Learning rate	[5e-5, 1e-4, 5e-3, 1e-3, 1e-2]	1e-3	5e-5

Table 24: Parameter settings for the Joint and Standalone models. "_" implies, parameter was not tuned or is not applicable for the respective model setup.

LCAM-BiLSTM (using a BiLSTM encoder) (b) LCAM-BioBERT (using BioBERT encoder), whereas the latter is achieved by fine-tuning the original (c) BioBERT and (d) SciBERT [18] models. The datasets are novel in the sense that the outcome type labels of the outcomes are drawn from Dodd et al. [55] taxonomy, which is not the basis of prior outcome annotations such as the EBM-NLP dataset.

LCAM-BioBERT outperforms standalone BioBERT models for both OSD and OC tasks.

	Task		OSD			OC		
Dataset	Model	setup	P	R	F	P	R	F
EBM-COMET	Baseline	Joint	63.0	55.0	59.0	78.0	73.0	74.0
	BioBERT	Standalone	74.0	74.3	**74.2**	76.7	78.4	77.5
	SCIBERT	Standalone	72.3	72.9	72.6	76.3	78.1	77.2
	LCAM-BioBERT	Joint	73.0	64.0	68.0	83.0	76.0	**83.0**
EBM-NLP$_{rev}$	Baseline	Joint	49.0	40.0	44.0	65.0	59.0	61.0
	BioBERT	Standalone	48.2	51.5	49.8	65.7	74.6	**69.9**
	SCIBERT	Standalone	48.5	49.7	49.1	64.2	66.5	65.3
	LCAM-BioBERT	Joint	57.0	49.0	**51.0**	67.0	65.0	66.0
EBM-COMET+EBM-NLP$_{rev}$	Baseline	Joint	62.0	54.0	58.0	68.0	64.0	65.0
	BioBERT	Standalone	58.6	61.4	60.0	81.4	83.0	**82.2**
	SCIBERT	Standalone	56.2	62.3	59.1	73.4	75.7	74.5
	LCAM-BioBERT	Joint	61.0	61.0	**61.0**	78.0	72.0	75.0

Table 25: Outcome span detection (OSD) and Outcome classification (OC) results in terms of F1 on the three datasets. Baseline, is a LCAM architecture with a BiLSTM sequence encoder.

5.4.3 Main Results

The first set of results reported in Table 25 are based on the independent test sets (Table 23) for each of the datasets. The joint LCAM-BioBERT and standalone BioBERT models are not only competitive

but they consistently outperform the baseline model for both OSD and OC tasks. I observe the LCAM-BioBERT model outperform the other models in the OSD experiments for the last two datasets in Table 25. On the other hand, the standalone BioBERT model achieves higher F1 scores for the last two datasets in the OC task.

5.4.3.1 Impact of the abstract context injection and Label attention

	LCAM	OSD(F)	OC(F)
EBM-COMET	- Attention	-10.0	-12.0
	- Abstract	-3.0	-5.0
EBM-NLP$_{\textbf{rev}}$	- Attention	-9.0	-7.0
	- Abstract	-7.0	-2.0
EBM-COMET	- Attention	-11.0	-15.0
+EBM-NLP$_{\textbf{rev}}$	- Abstract	-3.0	-1.0

Table 26: OSD and OC performance percentage decline when either the attention mechanism or the abstract representation are eliminated from the joint learning model (LCAM-BioBERT).

As shown in Table 26, the performance deteriorates (with respect to the results reported in Table 25) without the attention layers ("- Attention") by averagely 10% for OSD and 11.3% for OC. Similarly, exclusion of the abstract representation ("- Abstract") leads to an average performance decline of 4.3% for OSD and 2.7% for OC. As observed the decline resulting from "- Abstract" is less significant than that resulting from "- Attention" for both OSD and OC tasks.

This decline explains the significant impact of both (1) the semantic relational information between both tokens and labels as well as outcome spans and labels gathered by the attention mechanism, (2) information from the text surrounding a token or an outcome span embedded into an abstract representation. This therefore justifies inclusion of both these components.

5.4.3.2 Impact of Aligning Comparable Datasets

Overall, this result shows that the proposed label alignment method enables us to improve performance for both OSD and OC tasks.

To evaluate the label alignment method proposed in Section 3.4, I train a model using the aligned dataset (EBM-COMET+EBM-NLP$_{\textbf{rev}}$) and evaluate it on the test sets of the original datasets, reporting results in Table 27. I obtain significant improvements in F-scores for OSD in both EBM-COMET and EBM-NLP$_{\textbf{rev}}$. Additionally, for OC, significant improvement in F-score on EBM-NLP dataset and a slight improvement in F-score on the EBM-COMET dataset is observed. Overall, this result shows that the proposed label alignment method enables us to improve performance for both OSD and OC tasks.

	OSD			OC		
LCAM-BioBERT	P	R	F	P	R	F
EBM-COMET	73.0/83.0	64.0/64.0	68.0/71.0	83.0/90.0	76.0/80.0	83.0/84.0
EBM-NLP$_{rev}$	57.0/60.0	49.0/47.0	51.0/53.0	65.0/76.0	65.0/72.0	64.0/74.0

Table 27: Effect of dataset merging via label alignment. For each dataset, I report the performance on its test split obtained by LCAM-BioBERT trained on the corresponding train split (shown on the left side of /) vs. on the merger of the train splits of EBM-COMET and EBM-NLP (shown on the right side of /).

5.4.4 LCAMs multi-label set performance

LCAM-BioBERT model outperforms other multi-label models such as label specific attention networks used as baselines in multi-label prediction performance.

To further evaluate the LCAM-BioBERT model, I focus on the OC task results alone where the classifier returns the outcome types given an outcome span, and compare MLP performance to the baseline and another related MLP model, label-specific attention network (LSAN) [229], that learns BiLSTM representations for multi-label classification of sentences. For comparison, I compute P@n and nDCG@n using formulas similar to [229]. As illustrated in Figure 9, the LCAM model outperforms its counterparts for all datasets, and most notably for P@1. The joint BiLSTM baseline model performs comparably with LSAN, and indeed outperforms it on the EBM-COMET dataset for P@1, nDCG@1 and nDCG@3.

I attribute LCAMs superior performance to (1) Using a domain-specific (biomedical) language representation model (BioBERT) at its encoding layer, (2) Applying label-specific attention prior to classifying a token as well as before classifying the mean pooled represen-

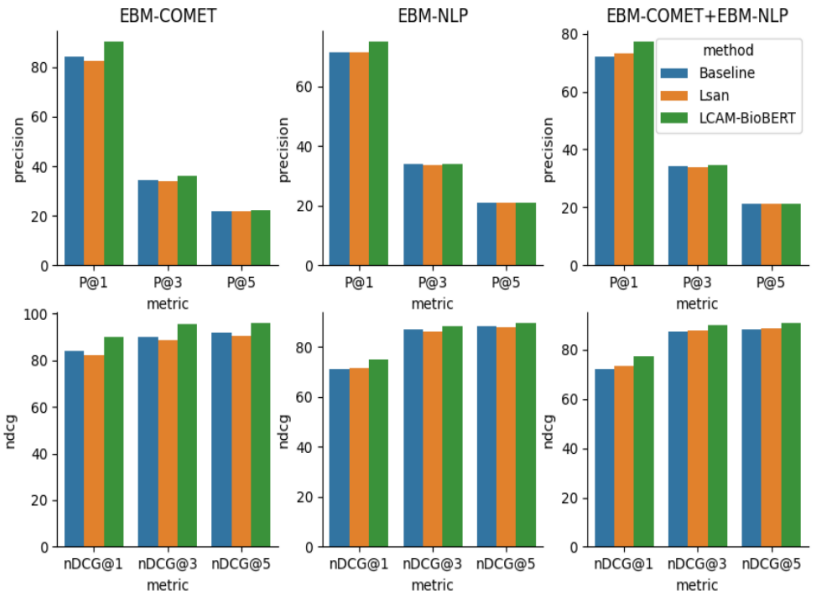


Figure 9: P@n and nDCG@n for three datasets

tation of an outcome span and finally (3) injecting global contextual knowledge from the abstract into the token and document (outcome-span) representations.

	Example Input sentence	Predicted labels P@1	Predicted labels P@2
Ground truth	The primary outcomes were **hospitalised death**[1], **severe disability**[2] at 15 months of age, **neonatal behavioural neurological**[3] assessment (nbna) score at 28 days of age, and Bayley scales of **infant development**[4] (BSID) score (including **mental development**[5] index (mdi) score and **psychomotor development**[6] index (pdi) score) at 15 months of age at follow-up.	1. Mortality 2. Life-Impact 3. Life-Impact 4. Life-Impact 5. Life-Impact 6. Life-Impact	
LCAM Output	The primary outcomes were hospitalised **death**[1], **severe**[2] **disability**[3] at 15 months of age, neonatal behavioural neurological assessment (nbna) score at 28 days of age, and Bayley scales of infant development (BSID) score (including mental **development**[4] index (mdi) score and **psychomotor development**[5] index (pdi) score) at 15 months of age at follow-up.	1. Mortality 2. Physiological 3. Life-Impact 4. Life-Impact 5. Life-Impact	
Ground truth	These results confirm retrospective studies and add that histopathology subtype is a strong determinant of **disease-free survival (DFS)**[1], in resected MAGE-A3-positive MSCLC.	1. Physiological	1. Mortality
LCAM Output	These results confirm retrospective studies and add that histopathology subtype is a strong determinant of **disease-free survival**[1] (DFS), in resected MAGE-A3-positive MSCLC.	1. Physiological	1. Mortality
Ground truth	The duration of total **hospital stay**[1], and **postoperative hospital stay**[2] in the ag (10.86 +/- 5.64, 5.69 +/- 4.55) d were significantly shorter than that in the cg (.10.86 +/- 5.64, 5.09 +/- 4.55) d (p=0.01, p=0.01))	1. Resource-use 2. Resource-use	
LCAM Output	The duration of total **hospital**[1] **stay**[2], and **postoperative**[3] **hospital stay**[4] in the ag (10.86 +/- 5.64, 5.69 +/- 4.55) d were significantly shorter than that in the cg (.10.86 +/- 5.64, 5.09 +/- 4.55) d (p=0.01, p=0.01))	1. Resource-use 2. Physiological 3. Physiological 4. Resource-use	

Table 28: Sample error predictions made by the joint learning model, with coloured words representing the outcome phrase (both in ground truth and output) and the colours representing different outcome types which are output. For multi-label predictions, I include P@1 and P@2 to indicate the top most predictions for the outcome phrase in question such as in example 2.

5.4.5 Error Analysis

I review a few sample instances that exhibit the mistakes the joint LCAM model makes in the OSD and OC tasks in Table 28.

OSD ERRORS: The model is observed partially detecting outcome phrases e.g. In Example 1, it detects death instead of hospitalised death, development instead of mental development, and in Example 2, it does not detect "(DFS)" as apart of the outcome phrase. Additionally, it completely misses some outcomes such as infant development in Example 1.

OC ERRORS: Incorrect token-level predictions will most likely result into incorrect outcome classification. In Example 1, Instead of se-

vere disability, the model detects "severe" as an outcome and "disability" as a separate outcome and classifies them as Physiological and Life-Impact respectively. Similarly, in Example 3, both outcomes are misclassified because at token level multiple outcomes are detected rather than one, hospital and stay rather than hospital stay, postoperative and hospital stay rather than postoperative hospital stay.

5.5 SINGULAR TYPE (LABEL) OUTCOME SPAN DETECTION (ST-OSD)

ST-OSD aims to detect and classify an entity span into one entity-type classification label. OSD described in the sections above achieved multi-label classification of outcome spans

To further investigate any performance gains or losses in the tasks, I re-formulate the OSD task into the conventional sequence labelling for NER where every entity is associated with at most one label [199]. Different from the multi-label classification achieved in OSD as described in earlier sections of this chapter, ST-OSD aims to detect and classify an outcome span into one outcome-type classification label. The main motivation for this investigation is that, the proportion of multi-labeled outcome span annotations is significantly smaller than that of singular labeled outcome spans, i.e. ca. 5% and 95% respectively of the total outcomes as shown in Table 29. For this conventional NER ST-OSD task, I discard the multi-labeled spans from the train and test sets and re-label the spans with BIO tags with their corresponding outcome types (Adverse-effects, Physiological, Life-impact, Resource-use, Mortality), i.e. the new label set $\mathcal{L}$ contains {B-Physiological, I-Physiological, ..., I-Mortality, O} where O represents non-outcome tokens. Given an input sentence $s = w_1, \ldots, w_M$, the goal of ST-OSD is to classify a token w_i into one of the labels in $\mathcal{L}$.

5.5.0.1 Model Architecture for ST-OSD

The seq2seq2set LCAM architecture illustrated by Figure 8 and described under Section 5.3, is retained minus the "set" component which performs OC by classifying a detected span into one of the outcome types. In this section we refer to the proposed architecture as LCAM − set since it excludes the text classification component, "set". Similar to LCAM, an input sentence is encoded by an encoder (a BiLSTM baseline or BioBERT), then these encoded representations are enriched with encoded representations of the abstract (from which

		EBM-COMET		EBM-NLP$_{rev}$		EBM-COMET+EBM-NLP$_{rev}$	
		Train	Test	Train	Test	Train	Test
Count	Multi-labeled	520	44	479	112	999	156
	Singular-labeled	2799	784	21197	5123	23996	5907
	Total	3319	828	21676	5235	24995	6063

Table 29: Statistics of multi-labeled and singular-labeled outcome span annotations in the investigated datasets

the sentence is extracted), a label attention mechanism is applied to make the tokens label-aware, after which a softmax layer is used to calculate the probability of a label given a tokens hidden state representation $P(\ell|h_i)$, where $\ell \in \mathcal{L}$. Similar to the setup in Table 24, I additionally fine-tune BioBERT and SciBERT for this ST-OSD task. In this architecture, only the token level loss i.e. L_{osd} as shown in Equation 33 is minimised.

5.5.0.2 Experimental results - OSD Vs ST-OSD

The model architecture is evaluated using the training and test sets used for the experiments whose results are reported in Table 25. Table 30 shows results of the OSD and OC task (achieved with joint learning as seen in Section 5.3.1) using multi-labeled outcome span annotations, and those of ST-OSD task which not only disregards the joint learning component but also discards the multi-labeled spans. Only experiments under ST-OSD are performed and none of the experiments under OSD and OC are repeated as they were already obtained and presented in Table 25. For comparison, both OSD and OC results from the earlier performed experiments are considered. For the ST-OSD experiments, I use an evaluation criteria that rewards models a prediction score of 1 for a prediction that matches both the

	Task		OSD			OC			ST-OSD		
Dataset	Model	setup	P	R	F	P	R	F	P	R	F
EBM-COMET	Baseline	Joint / LCAM*	63.0	55.0	59.0	78.0	73.0	**74.0 /**	57.0	55.2	56.1
	BioBERT	Standalone	74.0	74.3	74.2	76.7	78.4	**77.5**	69.4	74.7	**71.9**
	SCIBERT	Standalone	72.3	72.9	72.6	76.3	78.1	**77.2**	72.1	71.7	71.9
	LCAM-BioBERT	Joint / LCAM*	73.5	61.3	68.0	83.0	76.0	**83.0 /**	60.0	76.0	66.8
EBM-NLP$_{rev}$	Baseline	Joint / LCAM*	49.0	40.0	44.0	65.0	59.0	**61.0 /**	33.8	41.0	37.1
	BioBERT	Standalone	48.2	51.5	49.8	65.7	74.6	**69.9**	44.9	48.5	46.6
	SCIBERT	Standalone	48.5	49.7	49.1	64.2	66.5	**65.3**	48.6	48.1	48.3
	LCAM-BioBERT	Joint / LCAM*	57.0	49.0	51.0	67.0	65.0	**66.0 /**	54.1	47.2	50.4
EBM-COMET+	Baseline	Joint / LCAM*	62.0	54.0	58.0	68.0	64.0	**65.0 /**	58.9	51.6	55.0
EBM-NLP$_{rev}$	BioBERT	Standalone	58.6	61.4	60.0	81.4	83.0	**82.2**	56.3	58.9	57.6
	SCIBERT	Standalone	56.2	62.3	59.0	73.4	75.7	**74.5**	51.7	60.5	55.8
	LCAM-BioBERT	Joint / LCAM*	61.0	61.0	61.0	78.0	72.0	**75.0 /**	57.3	58.9	58.1
Average					58.8			**72.6**			56.3

Table 30: Table comparing results of OSD and OC targeting multi-labelled outcome span annotations obtained earlier (Table 25) with ST-OSD targeting singular-labeled outcome span annotation introduced under Section 5.5. Results of the Joint setup achieving both OSD and OC are separated from LCAM* (which indicates LCAM-set i.e. eliminates the set component from LCAM) by a slash. A BiLSTM is used as a baseline. Boldened and underlined results are the best and second-best F1 scores respectively in a single row e.g. For the EBM-COMET, 74.0 and 59.0 are the best and second best F1's obtained using the Baseline for OC and OSD respectively.

Better performance results for the OSD task in contrast to the ST-OSD investigated in this section

exact boundary surface string and the outcome type or label given the ground truth.

In this investigation, the best performance results are observed in the OC task in contrast to OSD and ST-OSD. Nonetheless, OSD results are second best outperforming ST-OSD. It is important to note the distinction across the output space (or label set size) for each of these three different tasks i.e. OSD label space only contains $B-Outcome, I-Outcome, O$ whereas OC label space contains the five outcome types (Adverse-effects, Physiological, Life-impact, Resource-use, Mortality). Both these label spaces are much smaller than $\mathcal{L}$ used in ST-OSD as elaborated in Section 5.5. Therefore, the performance decline observed in ST-OSD is attributable to 1) the increased dimension of the label space which introduces prediction difficulty because the cross entropy loss function is required to enumerate across all possible outputs [31, 42] 2) Eliminating the joint learning which implies that the token level classification model (achieving OSD) would no longer benefit from learning the compatibility constraints or correlations between token- and sentence-level (spans outcome types or labels) which is relevant in jointly classifying token and outcome span representations and 3) the skewed label distribution with some outcome type labels having fewer training instances than others [31] as seen in Table 31.

Additionally, I observe LCAM* (LCAM-set) outperform the the standalone setup, although BioBERT and SciBERT achieve better performance on the EBM-COMET. The superiority of LCAM-set is attributed to the abstract context injection and the label attention that are used in the LCAM architecture to enrich the tokens with global contextual information and label aware information. Overall, the multi-label OC achieves the best performance as shown in the average results row, which further suggests that NER tasks can benefit from joint modelling such as what is achieved in the OSD task.

	Adverse-effects	Mortality	Life-Impact	Resource-use	Physiological
EBM-COMET	745	54	117	139	3092
EBM-NLP$_{\textbf{rev}}$	1480	875	3814	3635	17107

Table 31: Frequency distribution of samples across outcome types or labels in EBM-COMET and EBM-NLP$_{\textbf{rev}}$

5.6 DISCUSSION AND SUMMARY

Given real-world scenarios where it is often impractical or computationally demanding to build a model for each and every single task, it is imperative to build models that can multi-task or simultaneously perform multiple different tasks. Multi-tasking in NLP has recently achieved outstanding success with the release of models that can si-

multaneously perform a diverse set of tasks such as MT, QA, Text Summarisation etc [174]. Inspired by the recent success MTL has achieved through unifying architectures specific to different tasks, this chapter has proposed and presented a method that combines and simultaneously achieves two different but related OD tasks of OSD and OC. OSD is concerned with detecting spans of tokens that correspond to or indicate health outcomes, whereas OC is concerned with classification of health outcomes into a set of predefined outcome types.

The chapter highlighted the correlation between OSD and OC as a key motivating factor in designing the joint learning model (LCAM) i.e. the token-level outcome span detected will influence the outcome type (sentence-level) assigned to the text span in which the outcome is mentioned and likewise, the outcome type classification must be consistent with the outcome span detected. In order to model this relationship, LCAM uses a label inclined attention to capture the relationships between words and their token-level labels as well as outcome spans and their outcome type labels. LCAM additionally augments token-level representations with contextual representations generated from the abstracts from which model input sentences are extracted.

I trained LCAM using BioBERT and SciBERT embeddings on both OSD and OC jointly, and observed performance gains over standalone (disjoint) setups in which the embedding models were each used to independently achieve OSD and OC. In the experimental results using EBM-COMET, EBM-NLP$_{\mathbf{rev}}$ datasets and the two merged (EBM-COMET+EBM-NLP$_{\mathbf{rev}}$), I observe LCAM-BioBERT improve the standalone performance by an average of 2.0% F1 in OSD tasks in both EBM-NLP and EBM-COMET+EBM-NLP$_{\mathbf{rev}}$. An even more significant improvement of 5.7% F1 is achieved in OC task on the EBM-COMET. An ablation analysis revealed that elimination of the abstract representation and the attention mechanism would hurt the performance of LCAM, thus justifying the inclusion of both components in the model's architecture. Using a common test set, I observed consistent improvement of F1 scores using the merged dataset EBM-COMET+EBM-NLP$_{\mathbf{rev}}$ across all tasks and both datasets, which validated the impact of the label alignment method proposed and used in merging the two datasets in Section 3.2. Furthermore, LCAM architecture outperformed counterpart MLP architectures in the OC tasks.

Various other entity recognition tasks that may not necessarily be within the clinical domain can benefit from this joint learning approach, particularly if the number of entity type labels is very large. However for the clinical domain, an example can be disease recognition and classification based on a hierarchical classification such as the human disease ontology, where a disease can be associated with not just a single fine-grain classification, but also the parent or super-class classifications in the ontology e.g. Down syndrome is a

chromosomal duplication syndrome, but also a chromosomal disease and a genetic disease.

This chapter and previous chapters have focused on enhancement of the retrieval of health outcomes from clinical text using PLMs. In the next chapter, the aim is to probe for the knowledge PLMs have about outcomes, the emphasis is shifted to investigate whether these PLMs can automatically answer questions querying for health outcomes.

6

POSITION-BASED PROMPTING FOR HEALTH OUTCOME GENERATION

6.1 INTRODUCTION

So far in this thesis, we have observed that LMs and particularly pre-trained CLMs can lead to substantial performance improvement in both OC tasks (as revealed in Chapter 3) and OSD (as shown in Chapter 4) as well as joint OSD and OC in Chapter 5. Inspired by Petroni et al. [170] who examined the factual knowledge memorization ability of LMs by exploring whether they can serve as an alternative to KBs, I investigate the extent to which LMs memorise health outcome related knowledge in this chapter.

LM-as-KB implies the usage of LMs as an alternative or atleast a proxy for explicit KBs

Language models (LMs) as knowledge bases (KBs) (LM-as-KB) is a rapidly growing phenomenon attracting a lot of attention in the NLP community [27, 170, 187, 190]. LM-as-KB implies the usage of LMs as an alternative or at least a proxy for explicit KBs. To achieve LM-as-KB, researchers adopt prompt-based learning PBL in which LMs learn to probabilistically predict missing information once given fill-in-the-blank prompt inputs [135] such as "Eiffel tower is located in ___". PBL has generally been a success, for example, in a systematic survey of prompting methods, Liu et al. [135] indicate that *"pre-train, prompt and predict"* is a new paradigm replacing *"pre-train and fine-tune"* paradigm in NLP. Because of this success, the rationale that LMs contain factual retrievable knowledge is ostensibly justified and therefore continually explored.

The prompt sequences often used in PBL have a masked token or span (denoted by [MASK] in the remainder of this chapter) that positionally appears either in the middle (Cloze-style) [49, 170, 187] or at the very end of the sequence (Prefix style) [173, 190]. Moreover, I learn that the majority of the PBL tasks probe relational knowledge possessed by PLMs [51, 94, 170], which implies that the prompt inputs used in querying the PLMs have to contain relational information such as *"subject-relation-object"* triples. Furthermore, I observe that, a fair amount of time in several PBL tasks is spent reconstructing prompt inputs through manually designing templates [51, 170] or corrupting prompt inputs through deletion [123], replacement [174] or permutation [75].

Prompt based learning involves probing for relational knowledge containing subject-relation-object triples possessed by PLMs

As discussed above, it is noticeable that, the syntactic and semantic structure of prompt inputs is a constraint encountered in PBL, notwithstanding the multitude of constraints that could arise given that PBL is inherently a text generation task [135]. This constraint

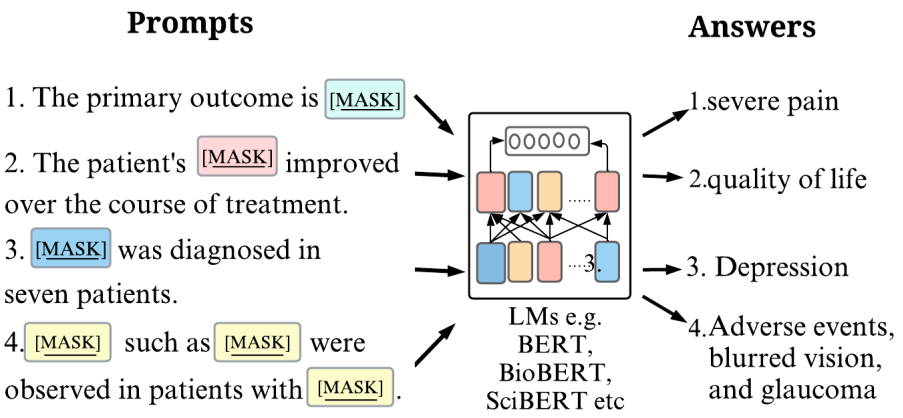


Figure 10: Prompt query variants used for probing evidence (in form of health outcomes) from PLMs, including common styles like Prefix (1) and Cloze (2) style, as well as rare styles Postfix (3) and Mixed (4) styles with [MASK] token/s at the beginning and in multiple positions in the prompt.

PBL often involves reformulating prompts to meet a particular linguistic pattern, however, it risks corrupting the grammar and the search space of possible patterns is almost unfathomable.

will usually require researchers to laboriously prepare supervised data with prompts whose linguistic patterns suit the objective of the prompting task. For instance, Davison, Feldman, and Rush [51], Heinzerling and Inui [75], and Jiang et al. [93] use templates that reformulate prompts to contain relational information connecting a particular text span to the to-be filled information. However, template-based prompt reformulation has two main challenges. First, it presents a risk of corrupting the grammar of the prompts unwittingly [51]. Second, the search space of the candidate prompts is too large [66] and is practically impossible to create templates that can enumerate all possible linguistic patterns that prompt queries can be tailored to. For example, prompt template patterns with missing information at the beginning and or with multiple missing information in a sequence are yet to be explored in prior works.

PBP shifts the emphasis off subject-relation-object triples to the [MASK]s positions as well as the interaction of all the other words with the [MASK]'s position

To address the above-mentioned challenges, I propose a strategy I denote Position-based Prompting (PBP), which is less concerned about the linguistic pattern or shape the prompt takes on, but rather focuses on the words (that the prompts are composed of) and their positions relative to the [MASK]. PBP is focused on shifting the emphasis on "*subject-relation-object*" triples to the masked positions as well as the interaction of all the other words with the [MASK]s position. PBP is built to automatically adjust from one prompt template to another, which essentially eliminates the need to prepare hand crafted prompts in the event that an LM is to be probed for rare knowledge. In its architecture, PBP enhances contextualised word representations with position-aware representations to solve fill-in-the-blank tasks. In this approach, I fine-tune PLM parameters along with position-oriented parameters to generate position-based contextualised word representations.

In this chapter, I investigate how well biomedical PLMs store and recall information relevant to health outcomes. In addition to the Prefix and Cloze styles, I incorporate two rare prompt style patterns that I denote Postfix and Mixed, where the former contains the [MASK] token/s at the beginning of the prompt sequence and the latter has multiple [MASK] token/s in various positions (Figure 10). My approach obtains mean scores (across several biomedical LMs) in Exact Match (EM) and Partial Match (PM) metrics that are an improvement (2.4% across both metrics) over those obtained using the vanilla PLM representations, reporting a significant improvement of 6.49% in F1 on the EBM-NLP [161] dataset. As later defined in Table 6.4.1, EM measures the percentage of predictions of all [MASK] tokens (or spans) that match the ground truth, whereas PM measures the percentage of correctly predicted [MASK] tokens.

6.2 ENTITY MEMORISATION AND RECALLING

Large-scale LMs with billions of parameters have already shown to recall facts that were observed in the training data [75, 93]. However, the ground truth for these LMs to achieve this is already laid with systematically handcrafted rules to follow in creating the prompt input sequences they receive at the training stage. For instance, the majority of the prompts created in PBL tasks embed knowledge in form of triples {*subject*,*relation*,*object*} such that LMs could correctly predict *object* entities when prompted with a sequence containing a *subject* and *relation* or otherwise predict *subject* entities when prompted with a sequence containing an *object* and a *relation* [93, 173, 200]. Whichever the case, models often predict answers as shown in (39).

LMs are capable of recalling facts they encountered during training, e.g. the can correctly recall an object if prompted with a subject and a relation or a subject if prompted with an object and a relation.

$$\hat{y}_i = \underset{y_i}{\text{argmax}} \; p([\text{MASK}] = y_i | x_{prompt}) \tag{39}$$

where i is the position of masked token within a prompt x_{prompt}.

This work however does not assume any prior knowledge contained in a prompt, but rather simply locates outcome entities in the sentences extracted from RCTs and mask them, an approach I refer to as *custom masking*. It is important to note that, this masking strategy slightly differs from the custom entity masking strategy in the Enhanced Language Representation with Informative Entities (ERNIE) [246]. In ERNIE, they randomly mask some tokens (within an input sentence) which would have been aligned to entities in a Knowledge graph. This arbitrary masking differs from the absolute target entity masking that is adopted in this chapter, where all entity mentions (outcomes) are masked.

6.3 POSITION BASED PROMPTING

In addition to formally defining the task I undertake, this section discusses the data used as well as the different stages of my proposed PBP strategy.

TASK FORMULATION: Let us consider an input prompt sequence s with one or more outcomes masked such that $s = x_1, \dots [M]_i \dots [M]_j \dots x_n$, where [M] is a masked token sequence, $[M] = \{x_i\}_{i \geqslant 1}^{i+|M|}$, $i \in [1, n]$ and $|M|$ is the length of the masked sequence. I consider four different prompt query variants shown in Figure 10: **Prefix prompts** contain [M] at the end of the prompt, **Cloze prompts** contains [M] in the middle of the prompt, **Postfix prompts** contain [M] at the start of the prompt, and **Mixed prompts** where there are several masked sequences distributed across the prompt. The questions I then pose are: (a) *can I determine how knowledgeable biomedical PLMs are of stored facts such as health outcomes?*, and (b) *If queried with any of the above variants, would these PLMs correctly fill in* [M]*s with the correct outcomes?*

DATASETS: Different from previous PBL works, I neither create custom templates nor do I reformulate prompts to follow an ideal linguistic pattern. I use plain raw sentences (that mention health outcomes) extracted from RCTs, which are contained in the revised version of EBM-NLP$_{\textbf{rev}}$ [1] and EBM-COMET [2] datasets. Extensive details entailed in constructing both datasets are included in Section 3.2 and Section 3.3. I do not eliminate any of the abstract sentences that do not mention outcomes, because I aim to familiarise the PLM (at fine-tuning) with text or context in RCTs which generally report about outcomes during clinical trial studies [225]. I refer to these sentences as *no_blank sequences* and use them alongside the prompt query variants introduced earlier. To my advantage, several sentence segments have no outcome annotations in both the EBM-NLP$_{\textbf{rev}}$ and EBM-COMET datasets.

I aim to familiarise the PLM (at fine-tuning) with text or context in RCTs which generally report about outcomes during clinical trial studies despite not containing outcomes.

6.3.1 Masked Language model and Prompt engineering

A hidden state h_i for each token in an input prompt s is extracted using a domain-specific PLM,

$$\mathbf{h_i} = \text{PLM}_\theta(x_i) \tag{40}$$

where $\mathbf{h_i}$ is a hidden state for the word x at position i. The matrix of hidden states for the entire input prompt is represented as $\mathbf{H} \in \mathbb{R}^{n \times k}$, where n is number of words in s and k is the hidden state size.

I define a function $f_{\textbf{prompt}}$ that concatenates the h_i in (40) to a randomly initialised d dimensional vector, which I denote as z_t corresponding to one of the four prompt query variants or the additional

no_blank sequences (introduced in Section 6.3), where $t \in$ [*prefix, cloze, postfix, mixed, no_blank*]. The function ensures that if an input s is a Prefix prompt, the corresponding vector z_{prefix} is concatenated to each h_i generated from s as shown in (41). This is done to enable knowledge transfer from one prompt query to another. For example, Mixed prompts are by construction a combination of Prefix, Postfix, and Cloze, hence they should benefit from information sharing via a common vector space.

$$f_{\textbf{prompt}}(h_i) = [z_t; h_i] \tag{41}$$

$z_t \in \mathbb{R}^{d_t}$, where z_t is a query type embedding of size d_t.

6.3.2 Position based conditioning (PBC)

To enrich the token representations, I propose a position-based attention mechanism to steer the model's focus on relevant information in the input prompt.

PBC uses an attention mechanism to incorporate position based information of the masked tokens relative to the non-masked tokens.

Positional ids in transformer-based architectures like BERT [54] are necessary for capturing the word order or the sequential nature of their input tokens. Without them, these LMs would not conveniently distinguish between two similar tokens in different positions (e.g. 'I' in "I like what I did"), which would limit their context-encoding ability. Unlike BERT's absolute position embeddings (APE) [216] which encode the position ids ordered from 0 to n-1, which are respectively indices of the first and last tokens in a sentence, the proposed relative position embeddings (PBC) encode the positions of each token relative to the distances to the masked tokens. In this approach, I use both positive and negative scalar values to distinguish between tokens that occur before and after the mask.

In this work, I define a sequence of position ids for each input prompt, where all masked positions take on an id of 0 and all the other tokens take id's relative to the masked position id. For example given a Cloze prompt with m tokens, I assign a mask at position i an id 0, and resulting sequence of position ids is $p = [1-i, 2-i, \ldots, -1, 0, 1, \ldots, (m-1)-i, m-i]$. This differs from the APE in BERT which would have been $p = [0, 1, 2, 3, \ldots, m]$, where $m+1$ is usually 512 for the maximum sequence length in BERT.

PBC uses an attention mechanism to incorporate position based information of the masked tokens relative to the non-masked tokens.

I compute an attention vector $A^{(s)}$, given by (42), for an input prompt s that allows each token to interact with every other token and retain knowledge of the relative position of the masked tokens in the input sequence.

$$\mathbf{A}^{(s)} = \text{softmax}(\mathbf{V}^\top \tanh(\mathbf{W}\mathbf{H}^\top + \mathbf{U}\mathbf{P}_s^\top)) \tag{42}$$

Here, $\mathbf{A}^{(s)} \in \mathbb{R}^{n \times 1}$, $\mathbf{V} \in \mathbb{R}^{k_a \times 1}$, k_a is size of attention layer, $\mathbf{W} \in \mathbb{R}^{k_a \times k}$, $\mathbf{P}_s \in \mathbb{R}^{n \times k_p}$ and $\mathbf{U} \in \mathbb{R}^{k_a \times k_p}$. $\mathbf{P}_s$ is a matrix of position

embeddings of size k_p extracted for each position p_n in the input prompt s. These embeddings are extracted from a trainable matrix $\mathbf{P} \in \mathbb{R}^{2n \times k_p}$ of randomly initialised vectors of size k_p for all possible positions $2n$ where n is the maximum sequence length, $|\{p_n\}_{-n}^{n-1}| = 2n$. The position based representation of each token is then computed with respect to the type of prompt. For the Prefix, Postfix and Cloze prompts, I obtain a prompt representation M^s given by (43).

$$\mathbf{M}^{(s)} = \mathbf{A}^{(s)}\mathbf{H} \tag{43}$$

Here, $\mathbf{M}^{(s)} \in \mathbb{R}^{n \times k}$. For the Mixed prompts in which there are multiple masked positions within the input sequence, I avoid biasing the attention mechanism towards masks at a specific position and thereby considering as many position id sequences as there are masked positions in the input prompt. For example, given a sequence with 3 masked positions, $s = [M], x_2, x_3, [M], x_5, x_6, [M]$, I obtain 3 position id sequences, i.e. the combined position id sequences is,

$$P^{(s)} = \bigcup_i P_i,$$

where each P_i is obtained with respect to the current mask position i. For the example above, $P^{(s)}$ = {[0,1,2,3,4,5,6], [-3,-2,-1,0,1,2,3,], [-6,-5,-4,-3,-2,-1,0]}, where the first position id sequence is obtained by treating the [M] at position 1, as mask at i, the second is obtained by treating the [M] at position 4 as mask at i and finally the third by treating [M] at the last position as mask at i. Attention vectors are computed for each position id sequence (P_i) and subsequently used to obtain the prompt representation $\mathbf{M}^s_{P_i}$. I compute the final representation of a Mixed prompt as the mean pool across these different representations,

$$\mathbf{M}^{(s)} = \sum_i^{|P^{(s)}|} \mathbf{M}^s_{P_i} \tag{44}$$

6.3.3 Prompt fine-tuning

The predicted probability of each vocabulary token is estimated via (45).

$$y = \text{softmax}(f(W_v \mathbf{M}^{(s)^\top}) \tag{45}$$

Therein, $W_v \in \mathbb{R}^{v^* \times k}$, v^* is the vocabulary size and f is a non-linear activation function. I use a BERT-based loss in predicting the masked tokens in each input given by (46).

$$\mathbf{L}_{PLM} = -\sum_{s \in \mathcal{T}} \sum_i^n \log P(y_i|s) \tag{46}$$

where $\mathcal{T}$ is the set of training example prompts. Some of the prompt query variants (Postfix and Prefix) are rare in the datasets, and some other prompt sequences are quite lengthy. This poses a challenge particularly when using small PLMs (with few parameters) to recall factual information. In order to mitigate model forgetfulness in such examples, I introduce an auxiliary task that computes a text classification loss as a cross entropy loss given by (47).

$$\mathbf{L}_{TC} = -\sum_{s \in \mathcal{T}} \sum_{i \in n} \log P(y_i | y_{<i}, s) \tag{47}$$

The overall training loss is defined as the weighted combination of the two losses as given in (48).

$$\mathbf{L} = \mathbf{L}_{PLM} + \lambda \mathbf{L}_{TC} \tag{48}$$

Similar to [44] and [186], I introduce a weighting parameter $\lambda (> 0)$ to adapt the auxiliary losses to the main mask prediction task.

6.3.3 Prediction

Similar to BERT [54], I consider generating outputs in parallel, initially treating the default representations provided by the model in (40) as a baseline and therefore use them to predict tokens in masked positions. I then use position-aware representation obtained using the attention mechanism in Section 6.3.2 to predict the mask tokens, calling these results Position-based conditioning (PBC). Lastly, I endeavour to retain the contextual knowledge presented by the PLMs as much as I possibly can by computing an average of the Baseline and PBC representations and term these Contextual PBC.

6.4 EVALUATION EXPERIMENTS AND RESULTS

In these experiments, I use several PLMs that are pre-trained on clinical texts such as PubMed abstracts, which often report outcomes such as BioBERT [121], SciBERT [18] and Biomed_RoBERTA [72]. Additionally, I include UmlsBERT because it augments BERT's pre-training input with semantic type embeddings aligned to clinical knowledge (semantic types) in the UMLS Metathesaurus [148]. I also use BERT [54] as a vanilla PLM that has not been pre-trained specifically on clinical texts.

6.4.1 Training and Evaluation

Unlike previous works where a particular relation within a prompt e.g. born-in, lives-in etc might appear multiple times within the train set, in this case, prompts are not semantically related in any way (i.e.

Because the prompts are not semantically related in anyway, I believe it might be harder for the model to memorise them, I therefore opt to train the models until the perplexity on the training data is low.

their is no relation knowledge that can be transferred over from one prompt to another). Because of the nature of the prompts, I believe it might be harder for the model to memorise them, I therefore opt to train the models until the perplexity on the training data reaches 1 or until the accuracy on the validation data saturates. I examine the model's generalisation ability to transfer knowledge to unseen prompts in few-shot and zero-shot settings. For the few-shot setting, I design experiments where I measure a model's accuracy in generating outcomes (as answers), which it encountered in a small number of prompts during training. The contexts in these evaluation prompts are not encountered during training. For example, consider an evaluation prompt – *"The patient's overall [MASK] improved according to the HRQOL questionnaire"*, the model would not have encountered the context surrounding the *"[MASK]"*. For the zero-shot evaluation, the model would have neither encountered the prompt nor the target outcomes during training. To simulate both the zero- and few-shot settings, I randomly split the datasets into train (80%) and test (20%) splits, and use the latter for the generalisation evaluation task shown in Table 35. I tune all hyperparameters using the validation data (20% of the EBM-COMET), and obtain optimal values as follows presented in Table 32.

METRICS: I define two different metrics for evaluating the proposed PBP strategy: Exact Match (EM) and Partial Match (PM). EM counts a prediction as 1 only if it matches completely with the correct answer, whereas PM uses the fraction of the overlapping tokens between the predicted and correct answers. Both EM and PM are averaged over all test instances to compute aggregated evaluation metrics, and therefore report their percentages in this work.

6.4.2 Results

In this section, I evaluate how well the model generates health outcomes when queried to answer a given prompt. For example, "*After patients were given sorafenib, they reported [MASK]*", the model should correctly generate the outcome *Fatigue* for the *[MASK]*.

Parameter	Tuned-range	Optimal
Train Batch size	[8,16,32]	16,32
Eval Batch size	[8,16,32]	8
Query type embedding size	[50,100,150]	50
Position embedding size	[100,200,300]	300
Attention layer size	[100,200,300]	200
Optimizer	[Adam, SGD]	Adam
Learning rate	[5e-5, 1e-4, 5e-3, 1e-3]	5e-5

Table 32: Parameter settings for the Position-based conditioning model.

Dataset-	EBM-COMET						EBM-NLP					
Method-	Baseline		PBC		Contextual PBC		Baseline		PBC		Contextual PBC	
Metric-	EM	PM	EM	PM	EM	PM	EM	PM	EM	PM	EM	PM
BERT	43.12	47.55	43.04	49.84	44.32	55.94	37.40	45.55	41.10	47.00	47.31	51.06
BioBERT	50.71	58.01	50.55	58.61	53.34	59.65	51.15	55.62	51.19	53.80	52.15	54.50
SciBERT	61.17	67.48	62.34	69.85	63.00	70.95	57.12	62.25	57.18	63.75	59.44	63.91
Biomed_RoBERTA	44.01	59.67	44.32	59.73	44.32	62.86	40.45	51.72	47.21	49.81	49.17	55.00
UmlsBERT	31.05	34.61	30.47	35.77	31.88	36.46	28.66	33.15	30.02	38.51	39.16	40.15
Mean score	46.01	53.46	46.14	54.76	47.37	57.17	42.96	49.66	45.34	50.57	49.45	52.92

Table 33: Table reports EM and PM accuracies of the various biomedical Pre-trained Language Models for the outcome recalling experiments. Mean score in a particular column is the average across all results in that column.

6.4.2.1 Outcome memorisation and retrieval

Table 33 shows the performance of the proposed PBC method in the outcome generation task. As observed, PBC consistently outperforms the baseline across most of the clinically informed BERT LMs (for both datasets), particularly for the PM results. More interestingly, I notice that Contextual PBC further improves the performance (both in EM and PM), indicating the importance of preserving the contexts in the position-based representations.

The proposed PBC method outperforms the baseline in the outcome generation task.

Comparing the different LMs, I found that, SciBERT performs best followed by Biomed_RoBERTA and BioBERT. Since all tested models follow the original BERT's architecture, I hypothesize that, the nature of corpora used in pre-training the best performing models was responsible for the performance, i.e. unlike UMLsBert and BERT, all the other models are pre-trained on text that includes PubMed abstracts, which often report outcomes. Additionally, I observe that PM results were generally better than EM results, which is attributable to the fact that PM is less strict compared to EM because it rewards the model for correctly generating a few of the tokens in the masked positions. Overall, the results suggest that PBC can be used to effectively retrieve facts such as health outcomes (biomedical entities) by simply augmenting contextual word representations with position-aware representations.

6.4.2.2 Prompt query variants

In Table 34, it is noticeable that the accuracy with which a model correctly answers Prefix prompts is significantly higher than that of the other prompts. I attribute this performance to the short length of these spans such as the one shown in Table 37 and the average number of tokens to decode per prompt. I also notice that the model struggles to correctly answer Mixed prompts compared to other types of prompts. I attribute this to the fact that, Mixed prompts are generally

The accuracy with which model correctly answers prefix prompts is higher than that of other prompts.

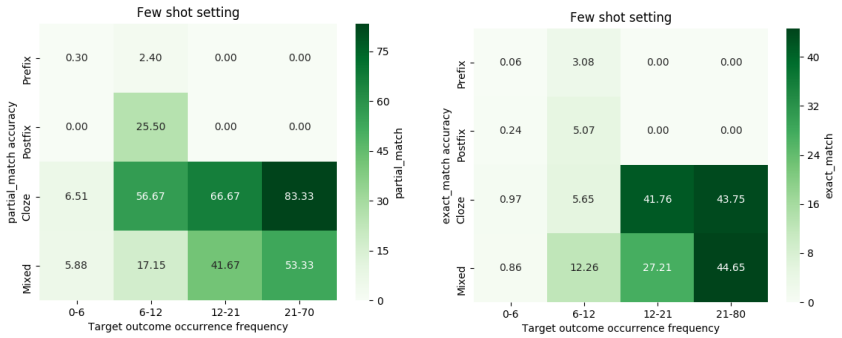
	#	Average prompt length	EM	PM
Postfix	65	18.5	48.43	58.51
Prefix	53	9.1	69.23	77.24
Cloze	630	24.2	50.08	60.49
Mixed	2594	38.8	43.68	45.46

Table 34: Exact Match (EM) and Partial Match (PM) accuracies for Outcome memorisation/recalling for the different prompt types using the EBM-COMET dataset.

	Cloze	Mix	Postfix	Prefix
#	174	613	13	12

Table 35: Number of prompts per prompt type used in evaluation of the few- and zero-shot settings.

very long sequences (38.8 tokens on average) and contain multiple masked positions to be predicted.



(a) Partial Match (b) Exact Match

Figure 11: Visualizing the Partial Match and Exact match accuracies when the best model (SciBERT+Contextual PBC+EBM-COMET) is trained with only a certain number of target outcomes.

To evaluate the model's generalisability, I fine-tune the model towards a small amount of target outcomes, and then measure the transferability of this knowledge by requiring the model to accurately generate these outcomes in prompts with completely different contexts.

6.4.3 Few- and Zero-shot Evaluations

To evaluate the model's generalisability, I fine-tune the model towards a small amount of target outcomes, and then measure the transferability of this knowledge by requiring the model to accurately generate these outcomes in prompts with completely different contexts. Test set prompts in Table 35 are carefully chosen using regular expression matching such that the contexts surrounding the missing outcomes are different from that of similar outcomes observed during training. For example, the model could have been trained on the outcome "adverse events" in five different prompts, and then at evaluation,

the model is required to generate the same outcome, however using prompts that are different from those encountered during training. By *different* here, I mean that the context (e.g. {ctxt} surrounding masks [M] in Table 37) in the prompt changes during this evaluation. Figure 11 plots shows results of model evaluation on prompts (Table 35). As observed in the plots, the model struggles to generate outcomes it hardly encountered during training (i.e. outcomes appearing in 0-6 prompts or 6-12 prompts). This is mostly evident in generating outcomes for Prefix and Postfix prompts, which is because there were not just few evaluated prompts of this types, but there were also few (53 and 65 respectively as shown in Table 34) in the train set. However, I see a trend of performance improvement when the frequency of target outcomes encountered during training increases, particularly for the Mixed and Cloze prompt.

6.5 ANALYSIS

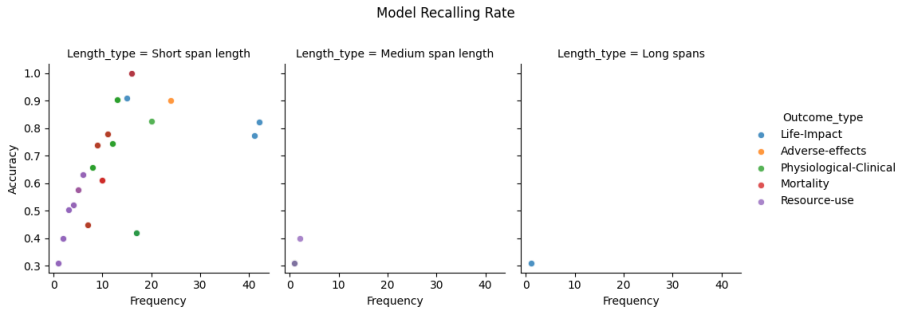


Figure 12: Analysis of the accuracy (PM) with which best model (SciBERT+Contextual PBC+EBM-COMET) recalls different types of factual information (outcome types) with varying span lengths and occurrence frequency (in the dataset).

There are more short span outcomes than they are medium span and long span outcomes. Within the short spans, I observe that the accuracy of recalling spans increases along with the frequency.

6.5.1 Impact of Length and Frequency of Outcomes

I partition the entire set of outcomes in EBM-COMET into 3 different groups based on lengths. Dividing the length of the longest outcome (22) by 3 returns approximately 7, which I use to create 3 groups i.e. 1) "short span length" to represent outcomes that are $\leqslant 7$ tokens long, 2) "medium span length" to represent outcomes of $7 >$ and $\leqslant 14$ tokens, and finally 3) "long spans" to represent outcomes of > 14 tokens long. Figure 12 shows how well the best model (SciBERT+Contextual PBC+EBM-COMET) performs when recalling outcomes of varying lengths and frequencies. Following prior work on EBM NLP, I endeavour to show the model's outcome recall rate by outcome type, which can be informative in terms of the complexity of modelling these outcomes. I firstly notice the skewed distribution of

outcome lengths with short spans dominant in the training sample. Unsurprisingly, I observe a trend of a performance increase as the frequency increases across the left hand plot with short outcomes, implying that the model struggles to recall infrequent outcomes despite their size but easily recalls the more frequent ones.

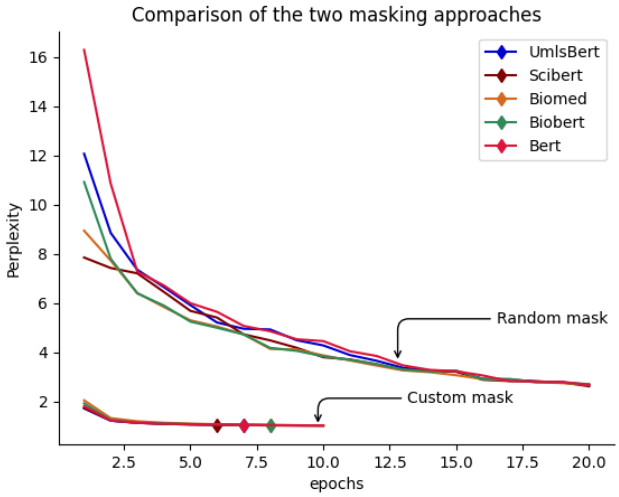


Figure 13: Achieving a target perplexity of 1.0 on the train dataset takes no fewer than 20 epochs with generic random masking of 15% of the input prompt tokens [54] compared to masking target factual information i.e. outcome spans themselves. Hitting target perplexity is shown using a diamond.

.

6.5.2 Random masking Vs custom masking

Figure 13 shows results of an ablation test in which I replace the custom masking approach with random masking. The key difference between the two is, while custom masking involves masking (or hiding) the outcomes in the prompts, random masking arbitrary masks 15% of the prompts tokens. As shown in the figure, the number of epochs required to reach a perplexity of 1.0 on the train data for the two masking approaches is almost incomparable, with custom masking quickly achieving this in approximately 7 epochs and random masking failing to achieve this, even after 20 epochs. The earliest random masking achieves 1.0 perplexity is 80 epochs for SciBERT, however I only visualise 20 epochs because of space. Besides this, the insight suggests that, custom masking would significantly reduce GPU runtime or otherwise minimise overwhelming computational resources with massive datasets.

Dataset	EBM-COMET				EBM-NLP			
Method	Contextual PBC (last layer)		Contextual PBC (Mean pool)		Contextual PBC (last layer)		Contextual PBC (Mean pool)	
Metric	EM	PM	EM	PM	EM	PM	EM	PM
BERT	44.32	55.94	45.80	57.19	47.31	51.06	47.45	53.41
BioBERT	53.34	59.65	53.58	61.22	52.15	54.50	54.80	55.15
SciBERT	63.00	70.95	63.15	72.67	59,44	63.91	60.08	66.93
Biomed_Roberta	44.32	62.86	45.00	63.17	49.17	55.00	49.19	56.33
UmlsBERT	31.88	36.46	33.10	39.21	39.16	40.15	41.12	42.41
Mean score	47.37	57.17	48.13	58.70	49.45	52.92	50.53	54.85

Table 36: Table reports EM and PM accuracies of the various biomedical Pretrained Language Models for the outcome recalling experiments using the EBM-COMET and Contextual PBC. Mean score in a particular column is the average across all results in that column.

6.5.3 Layer probing

I however explore an option of extracting a weighted average of representation across all layers

Initially, the hidden state used in (40) extracted from the last layer for each of the Biomedical PLMs for all experiments. I however explore an option of extracting a weighted average of representation across all layers (49) as a hidden state and study the performance of the models once this hidden state is introduced in the Position based conditioning framework to obtain position-aware representations.

$$\mathbf{h}_i^l = \text{PLM}_\theta(x_i) \tag{49}$$

$$\mathbf{h}_i = \text{MeanPool}(\mathbf{h}_i^1, .., \mathbf{h}_i^l, .., \mathbf{h}_i^{l_N}) \tag{50}$$

where h_i^l is a hidden state extracted from the l^{th} layer for word x at position i.

I only repeat training experiments using the Contextual PBC setup (Equation 6.3.3) however this time round using a mean pooled embedding across all layers as the hidden state. I notice in Table 36 that, aggregating a tokens representation by mean pooling across all layers of the transformer-based models does improve the performance in the outcome recalling experiments for both datasets.

6.5.4 Error Analysis

I analyse the outcomes generated by the best model (SciBERT+Cont extual PBC+EBM-COMET) during the few shot evaluation and notice that whilst the model generates correct outcomes for some prompts, it makes various kinds of mistakes. Table 37 includes a fair sample of

Query Variant	Prompt	Correct	Generated outcomes
Cloze {ctxt} [M] {ctxt}	Self-reported life-time medical diagnosis of **[M]** or use of antidepressants was considered as outcome.	- Depression	- Depression
Postfix [M] {ctxt}	**[M]** was assessed by questionnaires EORTC QLQ-C30, and EORTC QLQ-BR23 at baseline, and at three, six, and nine months.	- Quality of life	- Life
Prefix {ctxt} [M]	Two CMZ patients and one morphine patient showed complete **[M]**.	- pain	- unwanted pain
Mixed {ctxt} [M] {ctxt} [M] {ctxt}	Further additional benefits are better **[M]** and shorter **[M]** compared with standard GVHD prophlaxis without ATLG.	- quality of life (QOL) - immunosuppressive treatment	- immunosuppressive treatment
	The incidence of postoperative **[M]**, **[M]**, **[M]** and **[M]** was similar between the groups	- nausea, - vomiting, - drowsiness, -headache	- anxiety, - depression

Table 37: Example prompts from the test set and their predicted or generated outcomes for the outcome generation task. The Query variant column indicates the type of prompt as well as the prompt structure where {ctxt} implies context which might appear before, after or either ends of a masked sequence span.

the most commonly discovered mistakes. **Incomplete outcomes**, such in the Postfix where instead of "Quality of life", the model generates "Life". **Outcomes with irrelevant information**, such as Prefix case where the models generates more than what's expected, "unwanted pain" instead of "pain". Finally, **wrong outcomes**, where the model generates completely unexpected outcomes such as the case in the Mixed prompts.

It is however important to note that, the analysis of the wrong and correct answers is pegged on the answer or label space of the dataset, which is the list of all tokens within the dataset. Moreover, the objective maximizes the accuracy on the gold annotations in the test set after fine-tuning the LMs. In as such, the possibility of the false positives (incorrect) answers (such as is the case in wrong outcomes shown in the last row in Table 37) being correct given the prompt premise cannot be ruled out, because the available gold standard is not complete or exhaustive. Although, the evaluation performed is restricted to the gold annotations in test set, it is possible to search through all possible assignments given our unconstrained answer search space using a beam search [79] or top k sampling strategy [64]. However, this would require either a manually created test set of optimal answers or at least domain expert evaluation before establishing that the LM is achieving ideal performance. Such a model as PBP is applicable for information extraction and or abstractive question answering systems which requires models to produce answers that are often not mere sub-strings of the context in the question [105].

6.6 DISCUSSION AND SUMMARY

This chapter assesses the possibility of ignoring the constraint of aligning prompts to specific linguistic patterns in prompting tasks

that aim to store knowledge in LMs that could later be retrieved or transferred for fact generation tasks. In experiments using clinical domain datasets (supporting EBM tasks), I show that the position-based attention implemented over contextualised LMs can improve the ability of PLMs to recall facts such as outcomes (biomedical entities) encountered during training. I further observed that, the proposed model is able to generalise across unseen prompts, performing considerably well for Cloze and Mixed (extremely rare in PBL tasks) prompts. With the obtained experimental results, despite not aligning the prompts to commonly followed linguistic patterns, I can positively answer the question posed in Section 6.3 by claiming that PLMs are knowledgeable of stored facts.

7 CONCLUSION

7.1 INTRODUCTION

With an overarching aim to guide the search for biomedical evidence of effective interventions, this thesis has empirically used NLP methods to enhance the explicit automatic identification of health outcomes from clinical text. Preliminary studies on the subject of outcome detection (OD) or identification were primarily focused on classification of sentences (in PubMed articles (RCTs)) that constitute information relevant to outcomes. Results of these studies were encouraging especially because they would reduce multiline RCTs to lists of sentences mentioning outcomes, however, detection of the explicit individual/granular outcomes remained to be seen or achieved. Progressively, a handful of recent works attempted to annotate corpora for individual outcomes such as EBM-NLP [161] corpus, in order to facilitate detection of individual outcomes. The challenge with annotations such as EBM-NLP, is that they did not adopt any standard classification system for outcomes or other PICO elements, but rather used arbitrary classifications aligned to MeSH[1], which can then easily lead to noisy annotations.

Thus far, the scarcity of publicly available corpora has been reported as a chief deterrent responsible for the limited attention from NLP research community for the subject of OD. The inconsistencies in outcome reporting across Clinical Trial studies simply further stifles the involvement of the NLP community in this subject.

To this end, this thesis has reviewed, studied and explored OD as applied to EBM, with a strong emphasis on adoption of NLP methods and algorithms to aid pertinent OD sub-tasks of OSD and OC

To this end, this thesis has reviewed, studied and explored OD as applied to EBM, with a strong emphasis on adoption of NLP methods and algorithms to aid pertinent OD sub-tasks of OSD and OC. The thesis developed and proposed various methods to enhance extraction of explicit mentions of outcomes (OSD) in clinical text as well as classification of outcomes into core outcome types (OC). The thesis inherits and builds upon recent developments pertaining to OD including EBM-NLP, a corpus annotated for PICO elements [161] and a standardised taxonomy of outcome classifications [55].

This chapter provides a summary of the work covered in the different chapters that fulfilled the objectives established in the introduction Section 1.4. It discusses the limitations encountered in the course of conducting the work the thesis covers in Section 7.3. Furthermore, the chapter supplements the thesis contributions with a variety of real-word application scenarios in which these contributions can be

1 https://www.nlm.nih.gov/mesh/meshhome.html

adapted (Section 7.4). Finally, the thesis discusses potential areas the work can be improved upon in Section 7.5.

7.2 THESIS SUMMARY

The thesis commenced with a preamble in Chapter 1 that motivated the main work it covers on the detection of outcomes from clinical text. Key to the detailed motivation is the critical intervention of computational methods such as NLP to optimise the search and retrieval of clinical facts from unstructured clinical text published in huge volumes at an unprecedented rate. Importantly, this preamble asserts that, automating OD would inevitably speed up access to the best available evidence necessary in delivery of optimal patient care. From an NLP perspective, OD initially involved classifying sentences in RCTs as outcomes-statements to imply that the sentence summarised consequences of an intervention. Later on, OD was cast a sequence labelling task to detect mentions of individual outcomes (OSD) and subsequently a classification task in which outcome spans are hierarchically mapped to standardised core outcome types (OC).

The preamble further highlighted the main challenges in OD which are largely responsible for the limited attention OD has received from the NLP community and computer science fraternity at large. These challenges include, the variability with which outcomes are reported across various RCTs, absence of standardised outcome classifications and the scarcity of publicly available corpora to support building OD algorithms. In its conclusion, the preamble outlined the objectives and contributions the thesis made to the research paradigm that embodies OD, EBM.

Chapter 2 delves into the history of EBM NLP, which encapsulates all applications of NLP techniques to extract PICO elements, which collectively forms the basis of clinical questions used when searching for evidence of an intervention's effectiveness in biomedical literature.

Chapter 2 delves into the history of EBM NLP, which encapsulates all applications of NLP techniques to extract PICO elements, which collectively for the basis of clinical questions used when searching for evidence of an intervention's effectiveness in biomedical literature. To begin with, the chapter provides a background on Transfer Learning (TL) and its recent success in BioNLP. This background mainly serves to motivate the adoption of PLMs in the methods that the thesis goes on to propose in the subsequent chapters. The chapter progresses on to review prior dataset construction efforts which most notably relied on the structured nature of RCTs to prepare labelled datasets, i.e. sentences under a section heading "PARTICIPANTS" were labelled "Participants" or "p", those under section heading "OUTCOMES" were labelled "Outcomes" or "O" etc. Most recently, datasets with more granular annotations of outcome spans have been built to facilitate OSD.

Subsequently, the chapter extensively discusses two broad task setups in NLP, Sentence level classification (SLC) which aims to predict the likely label for a given sentence and Token level classification

(TLC) which aims to predict the likely label for a given single token. Additionally, the chapter explores prior attempts that have combined these two setups to simultaneously achieve multiple tasks such as joint DNER and DNEN. The chapter reviews prior work on two areas that drive some of the proposals the thesis makes. These areas include, noise reduction in distantly supervised as well as crowd sourced datasets and prompt based learning for text generation in BioNLP.

Chapter 3 addresses the first objective of the thesis which aims to evaluate and improve the reliability of current outcome annotations. The chapter implements a hybrid noise filtering framework that combines POS tagging and rule-based chunking to denoise flawed outcome annotation spans in a crowdsourced corpus, EBM-NLP. The framework uses a collection of heuristics that rely on lexical and syntactic features to filter out noise from annotated data. Each heuristic is strategically created to filter out specific noise (flaw), however correction of each flaw is not necessarily limited to a single heuristic. Experiments targeting OC showed that the proposed framework led to an improvement in the F1 classification scores for each outcome type.

Chapter 3 addresses the first objective of the thesis which aims to evaluate and improve the reliability of current outcome annotations.

The chapter also introduced and presented EBM-COMET, a dataset of PubMed abstracts expertly curated for the task of OSD. This dataset is distinct from earlier efforts in such a way that, outcomes are annotated at a granular level (spans of tokens in an outcome phrase) rather than at coarse level (sentences mentioning outcomes). More so, it uses standardised outcome classification labels drawn from a recently proposed taxonomy of standardised outcome classifications [55]. I used the dataset in fine-tuning a variety of PLMs and empirically showed that its annotations lead to an improvement in performance as well as faster convergence on the OSD task.

The chapter additionally proposed a label denoising approach that aims to automatically correct weak labels in the EBM-NLP corpus by replacing them with standardised outcome classification labels drawn from the outcome taxonomy proposed by Dodd et al. [55]. The denoiser is a flexible, re-usable unsupervised text alignment approach which extracts parallel annotations from comparable datasets, where one of the datasets is considered to have the standardised target labels. Experiments showed that, denoising EBM-NLP labels using this alignment approach led to significant gains in the F1 score of both OSD and OC.

The aim in Chapter 4 is to assess and advance OD tasks to new SOTA performance on benchmark datasets as established by the second objective. Using a comprehensive comparative assessment of fine-tuning and feature based TL adaptation methods, the chapter reached a consensus on which CLMs are suitable for OSD and OC, thereby nominating BioBERT and SciBERT. The fine-tuned BioBERT model outper-

formed prior SOTA results published by Brockmeier et al. [26] in PIO extraction (detecting spans of P, I and O elements). An additional contribution of this chapter is, it reveals the struggles that the fine-tuned models have in detecting full mention of outcome spans i.e. the accuracy with which the models detect full mention of granular outcome spans is lower than the accuracy with which they detect tokens that form outcome spans. The chapter also empirically showed that, the longer an outcome span, the more difficulty in detecting the outcome span i.e. the accuracy decreases with an increase in the lengths of outcome spans.

Chapter 5 fulfills objective 3 of the thesis by proposing LCAM, a joint learning framework that adopts pre-trained CLMs to simultaneously perform OSD and OC.

Chapter 5 fulfills objective 3 of the thesis by proposing LCAM, a joint learning framework that adopts pre-trained CLMs to simultaneously extract outcome spans (OSD) and classify outcome spans (OC). Unlike prior strategies that heavily rely on hand-crafted feature engineering, the proposed LCAM model leverages the compatibility constraints between the outcome spans and outcome types to achieve joint learning of a pair of tasks. An outcome type predicted for a text span in a sentence must be consistent with the other outcome spans detected from the same sentence, while the outcome spans detected from a sentence must be compatible with their outcome types. LCAM augments token-level representations with contextual representations generated from the abstracts from which model input sentences are extracted. Additionally, LCAM uses an attention mechanism that allows words at token-level to interact with the outcome type labels in order to generate a label context attention based representation. In my experiments, I observe that LCAM-BioBERT (LCAM consuming BioBERT embeddings) improves standalone performance (in disjoint setups) of BioBERT in both OSD and OC.

To verify and justify the evaluation performances obtained in the preceding chapters as required by objective 4 of the thesis, Chapter 6 investigates how knowledgeable biomedical LMs are of health outcomes. Having successfully used these biomedical LMs to extract and classify outcomes expressed in clinical text in Chapter 3 to Chapter 5, I sought to explore whether I can leverage the knowledge in these LMs for outcome generation. This chapter's work is motivated by prior efforts that explores the utility of LMs as KBs. Specifically, I propose a position-based prompting (PBP) framework that uses prompts to query LMs for health outcomes. Unlike previous works that heavily relied on constructing prompt templates that embed relational knowledge and are aligned to a specific linguistic pattern, my approach simply translates a given sentence mentioning an outcome/s into a fill-in-the-blank prompt with the outcome masked, and relies on a PLM to generate the outcome.

Chapter 6 investigates how knowledgeable biomedical LMs are of health outcomes.

The PBP framework uses an attention mechanism that captures information of each words position relative to the positions of the masked outcomes (outcomes to be generated) within the prompt. Be-

cause the approach ignores the constraint of aligning prompts to specific linguistic patterns, I introduce a new set of prompt templates (which to my knowledge have been understudied in prior work), these are, Postfix style templates and Mixed style templates which respectively have masked word at the front and in multiple random places including the front, middle and end of the prompt. In the experiments using EBM-NLP and EBM-COMET, I show that the the proposed framework improves the ability of PLMs to generate outcomes encountered during training. I further observe this framework generalise across unseen prompts, performing relatively well for Cloze and Mixed (extremely rare in PBL tasks) prompts. With the obtained experimental results, I can emphatically claim that, LMs memorise health outcomes they encounter during training, hence explaining their success in OSD as witnessed in preceding chapters.

7.3 LIMITATIONS

Entailed under this section are two main limitations encountered as I undertook the different tasks.

LIMITED PUBLICLY AVAILABLE ANNOTATED CORPORA: The challenges of obtaining high-quality training data are well documented by multiple authors across the NLP research community [24, 70, 161]. Most notably, the cost and the difficulty of the annotation exercise are two bottlenecks that recur as obstacles to curating quality supervised datasets for many researchers. In certain specialised domains such as medicine, both cost and difficulty can exponentially increase because of the nature of task and expertise required to complete the task [76]. The dearth of publicly available annotated corpora to facilitate OSD and OC was a main limitation in the survey and analysis performed in this thesis. All dataset construction efforts that predate the release of EBM-NLP corpus [161] and the EBM-COMET were not simply annotated at sentence level, but they were never annotated with core outcome types i.e. RCTs sentences are labelled with a tag such as "O" or "Outcomes" to indicate the sentence mentions an outcome. Intuitively, having a collection of datasets expertly annotated for outcomes at token-level would enhance my comprehensive analysis and provide multiple benchmark performances for future OSD and EBM related tasks.

The dearth of publicly available annotated corpora to facilitate OSD and OC was a main limitation in the survey and analysis performed in this thesis.

OUTCOMES ARE LOW-RESOURCED BIOMEDICAL ENTITIES: Despite the extensive analysis I perform, all the analysis is performed using supervised datasets whose primary purpose was to facilitate training of neural based architectures. While biological entities such as diseases, genes, DNA, proteins, chemicals etc enjoy a wealth of resources external to biomedical literature such as taxonomies, ontolo-

gies and databases[2] which can substantially support their analysis, health outcomes have none available. Performance of several existing NLP tasks has been improved by incorporating external knowledge accessible in relatable knowledge bases KBs.

COMPUTATIONAL COST: The series of Large LMs that have recently been released [27, 102, 174, 194] have demonstrated impressive performance on not just many downstream NLP tasks, but even in hard-pressed evaluation settings such as few-shot and zero-shot fine-tuning. The challenge though is, increasing the models size (number of model parameters) necessitates increasing the compute and energy cost [78]. Kaplan et al. [102] suggests that, a 5.5× increase in model size commands a 10× increase in computational budget. Fine-tuning these Large LMs would have potentially led to even better performance gains for the tasks the thesis undertakes, however Fine-tuning 110M parameters of BERT_{base} was already a heavy GPU intensive job for a 24G TITAN RTX GPU used for work in this thesis. Training BioBERT 768-dimensional word embeddings and randomly initialized Part Of Speech POS embeddings in a feature based TL approach using a BiLSTM required a combined total of 4-days ca. 96 GPU hours for all the datasets. Replacing BioBERT [121] with BioELMO (3072 dimensional embedding) [97] consumed even more GPU runtime of up to 13 days ca. 312 GPU hours. To address this, I used PCA dimensionality reduction to retain a smaller number of dimensions to fine-tune as discussed in Table 4.3.2.3. Overall, lack of more compute power was a limitation in adopting larger LMs for the OD tasks, and hence potentially missed achieving further performance gains as recent research using large LMs has proven [78].

7.4 RESEARCH APPLICABILITY

This section covers a summary of how the methods and approaches proposed in this thesis can be used within the healthcare systems.

CLINICAL RESEARCH QUESTION ANSWERING SYSTEMS: The rapid increase in biomedical literature available for clinical analysis has directly increased the need for efficient and effective tools to access and analyse biomedical literature [114]. In reviewing medical research evidence, clinicians are faced with a lot of clinical questions whose answers they need to access with less effort, cost and time [53]. Various digital archives for biomedical data such as PubMed and Clinical trials registry[3] have inbuilt search engines and information retrieval systems that can ably provide an array of results to clinicians, however, these results are often at coarse level i.e. it is a set of source articles

2 https://www.nlm.nih.gov/research/umls/index.html
3 https://www.clinicaltrials.gov/

or article sentences relevant to a search query [28, 53]. The challenge with this is that, clinicians then have to read through these multiple retrieved sources to locate more refined and granular information before they can satisfactorily answer the clinical question. Even though, these systems have more sophisticated features that can rank sources based on their relevance to a particular query, the required level of precision when answering clinical questions still remains unachieved [53, 165].

Incorporating NLP techniques such as the ones proposde in this thesis into clinical QA systems tailored for EBM would likelyimprove the results retrievable during clinical analysis

Incorporating NLP techniques such as the ones proposed in this thesis into clinical QA systems tailored for EBM would likely improve results retrievable during clinical analysis. Given a PICO formatted question that aims to discover the Outcome (O) component, a feature implementing the LCAM model (this thesis proposes) that achieves joint OSD and OC would not only retrieve the actual granular outcome from the text but also provide details of the type or domain the outcome belongs. For systems that already provide candidate article sources or sentences as answers to a PICO question, LCAM can enhance this result by performing further analysis on the candidate answers to provide a more refined answer hence reducing the effort required by the clinician to manually traverse the results.

SYSTEMATIC REVIEW ANALYSIS: In EBM, systematic reviews identify, assess, synthesize and interpret published and unpublished evidence for purposes of decision-making for clinicians, patients, policy makers and other stakeholders [88, 181]. Jonnalagadda, Goyal, and Huffman [100] use a survey to prove that automation of the data extraction step in a systematic review can substantially reduce the time necessary to complete and update a systematic review. The same author indicates in their survey that, data extraction can consume 2.5 to 6.5 years during system review analysis. BioNLP community has made positive strides to this effect, introducing techniques that automate the retrieval of text passages in form of sentences that contain evidence necessary of optimal patient care [22, 23, 95, 107].

Prior efforts have managed to achieve classification of sentences into one of the four PICO elements, and of late, some authors have built ML systems that extract spans of text corresponding to the PICO elements. These works have been a building block for the work described in this thesis, and even further enhancing the analysis and improving the performance of span extraction with specific focus on health outcomes. Differently from prior datasets that were annotated using arbitrary labels aligned to MeSH, this work publishes a dataset that uses standard outcome classifications drawn from a recent taxonomy of outcomes built from a rigorous study of trial registry entries,[4]

4 https://www.clinicaltrials.gov/

Cochrane Reviews[5] and Core outcome sets in COMET.[6] Some of my findings such as nominating effective biomedical PLMs for analysis of health outcomes can guide systematic review analysis that applies NLP methods. Moreover, I propose LCAM, a framework that is dynamic enough to perform multiple tasks including OSD and OC, both of which can enable clinicians have multi-faceted views of evidence when making decisions.

CLINICAL DIAGNOSIS: Automatic clinical diagnosis has received more attention in the recent past due to the recent advances in deep reinforcement learning [221]. Recent studies have proposed dialog systems which converse with patients to collect symptoms from them for automatic diagnosis [142, 221]. Training on datasets with annotated symptoms and diseases enables these systems to perform considerably well during evaluation.

In this work, I proposed a new task: health outcome generation that I tackle using prompt based learning PBL, in which LMs learn to predict missing information once given fill-in-the-blank prompt inputs such as "*Bill, middle-aged relatively tall skinny male adult suffered a fall and therefore likely outcomes are* ___. The proposed Position based prompting framework demonstrates an ability to automatically generate outcomes even when they were never encountered when fine-tuning it on a small set of prompts in a few-shot setting.

Motivated by the growing research on automated diagnosis tools briefly described in above paragraph, I envisage adoption of my prompt based system in tasks to auto-generate or auto completing diagnostics information such as signs and symptoms if provided some information in a statement such as those written in clinic letters by clinicians, most especially if all relevant details are specified in the statement. A potential use-case of such an application is "Automatic outcome span generation in a clinician-patient dialogue system" described below,

The input into the dialog system is a statement of free natural language text in which a clinician such as a General Practitioner (G.P) has specified a patient's age, problem and events during or following an illness, accident or health setback. Based on this statement, the prompt based system would then generate one or more outcomes that are potential symptoms the patient might be suffering.

CLINICAL INFORMATION EXTRACTION APPLICATIONS: The numerous potential applications of information extraction (IE) have attracted various technology and healthcare stakeholders. As applied to NLP, IE in the clinical domain involves automatically searching and retrieving concepts, entities, and events, as well as their relations and associated attributes from free text [218]. Developed by various

5 https://www.cochranelibrary.com/
6 https://www.comet-initiative.org/

stakeholders (most notably University College London Hospitals), CogStack[7] is an application framework that is enhancing automatic searching and extraction of specific clinical terms and data relevant to answering clinical queries such as "*Has the patient received any high cost treatments that have not been captured in their discharge summaries?*". Similar to CogStack, Amazon Comprehend Medical[8] (developed by Amazon) is an API that enables quick and accurate extraction of information such as medical conditions, medications, dosages, tests, treatments, Protected Health Information (PHI) etc from unstructured clinical text. Additionally, the API can identify relationships between extracted entities.

The clinical IE tasks that this thesis primarily focuses on (i.e. OSD and OC) are directly related to the usecases of both CogStack and Amazon Comprehend Medical applications. The correlation between the two suggests that, the various assessments and methods we propose are relevant in the processing pipeline of not just these two application, but various other applications. The denoising framework Section 3.2.2 and the text alignment approach Section 3.4 are potentially useful in pre-processing the unstructured text consumed clinical IE systems such as the aforementioned applications. Furthermore, the custom NER models the thesis proposes in Chapter 4 to achieve automatic detection of entities (health outcomes) from clinical trial abstracts can potentially complement features provided by CogStack and Amazon Comprehend Medical.

7.5 FUTURE WORK

Various methods, approaches and analysis were conducted and documented in this thesis, all in an effort to tackle and advance the task of extracting outcomes in clinical text from an NLP standpoint. From a research point of view, there are several open problems that future research can focus on to discover more effective and reliable ways of addressing OSD and OC for EBM. In this section, I provide a perspective of two potential future directions in this regard which mainly focus on probing for effective knowledge representation and how it can enhance existing LMs which were heavily used in the work in this thesis.

7.5.1 Task-adaptive pre-training for Outcome Span Detection

Further pre-training of a PLM on a task-relevant corpus of unlabelled text (also known as (TAPT)) has shown to be effective [72, 80]. Gururangan et al. [72] empirically showed that both domain-adaptive pre-training (DAPT) (further pre-training a PLM on unlabelled text of

7 https://cogstack.org/
8 https://aws.amazon.com/comprehend/medical/

a domain) and TAPT improve performance of the PLM on tasks specific to domains that included Biomedical, Computer Science, News and Reviews. Moreover, the same author showed in an extended analysis that increasing the task-specific data to further pre-train a PLM on leads to significant benefits in performance on the different tasks.

Motivated by Gururangan et al. [72], I envisage TAPT for OSD and OC would most probably improve the current performance obtained in this work. Most of the PLMs adopted in this work are pre-trained on corpora that is a mixture of general domain (such as news and wikipedia) and biomedical (such as PubMed) unlabelled text. In the future, it would be ideal to tune parameters of these PLMs on data specific to PICO elements or health outcomes using pre-training objectives such as MLM defined in Section 2.2.

7.5.2 Knowledge-enhanced Outcome Detection

There is an upsurge in interest in incorporating external knowledge into LM to solve both down stream and domain-specific tasks within the NLP community [168, 241, 246]. Yu et al. [240] surveys a variety of methods that have successfully been used to integrate knowledge into LMs, including model architectures like attention mechanism and graph neural networks, internal sources like topics and keywords and external sources like knowledge bases (KBs) and knowledge graphs (KG). Yuan et al. [241] perform text-entity fusion encoding in which they augment an entity's Transformer [213] encoded representation by adding to it a linked UMLS entity representation extracted from a KG. Their entity linking process involved searching for k nearest entities in a UMLS KG and computing a weighted sum of embeddings of these near embeddings to represent the linked entity representation.

Motivated by recent success in knowledge-enhanced language modelling, I envisage that more effort can be spent learning how to combine knowledge from different and diverse sources to improve representations for outcomes. Whereas popular knowledge bases with millions of entities and relation triples like UMLS do not have explicit knowledge about health outcomes, a list of resources (limited) such as COMET have knowledge relatable to outcomes either directly or indirectly. A potential ideal source would be one with descriptions of outcome domains and classification, which descriptions can be used in augmenting representations for outcomes.

BIBLIOGRAPHY

[1] Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. "Correcting crowdsourced annotations to improve detection of outcome types in evidence based medicine." In: *CEUR Workshop Proceedings*. Vol. 2429. 2019, pp. 1–5. URL: http://ceur-ws.org/Vol-2429/paper1.pdf.

[2] Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. "Assessment of contextualised representations in detecting outcome phrases in clinical trials." In: *European Journal of Biomedical Informatics* 17.9 (Aug. 2021). URL: https://arxiv.org/pdf/2203.03547.pdf.

[3] Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. "Detect and Classify – Joint Span Detection and Classification for Health Outcomes." In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 8709–8721. DOI: 10.18653/v1/2021.emnlp-main.686. URL: https://aclanthology.org/2021.emnlp-main.686.

[4] Salah Abdel-Aleem and Salah Abdel-aleem. *Design, execution, and management of medical device clinical trials*. Wiley Online Library, 2009. Chap. GLOSSARY OF CLINICAL TRIAL AND STATISTICAL TERMS, pp. 235–247.

[5] Lada A Adamic, Dennis Wilkinson, Bernardo A Huberman, and Eytan Adar. "A literature based method for identifying gene-disease connections." In: *Proceedings. IEEE Computer Society Bioinformatics Conference*. IEEE. 2002, pp. 109–117.

[6] Akiko Aizawa. "An information-theoretic perspective of tf–idf measures." In: *Information Processing & Management* 39.1 (2003), pp. 45–65.

[7] Alan Akbik, Duncan Blythe, and Roland Vollgraf. "Contextual string embeddings for sequence labeling." In: *Proceedings of the 27th international conference on computational linguistics*. 2018, pp. 1638–1649.

[8] Betty van Aken, Jens-Michalis Papaioannou, Manuel Mayrdorfer, Klemens Budde, Felix Gers, and Alexander Loeser. "Clinical Outcome Prediction from Admission Notes using Self-Supervised Knowledge Integration." In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Apr. 2021, pp. 881–893.

[9] Kamal Al-Sabahi, Zhang Zuping, and Mohammed Nadher. "A hierarchical structured self-attentive model for extractive document summarization (HSSAS)." In: *IEEE Access* 6 (2018), pp. 24205–24212.

[10] Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. "Publicly Available Clinical BERT Embeddings." In: *Proceedings of the 2nd Clinical Natural Language Processing Workshop*. June 2019, pp. 72–78.

[11] Jacopo Amidei, Paul Piwek, and Alistair Willis. "Identifying annotator bias: a new irt-based method for bias identification." In: *Proceedings of the 28th International Conference on Computational Linguistics*. 2020, pp. 4787–4797.

[12] Saadullah Amin, Katherine Ann Dunfield, Anna Vechkaeva, and Guenter Neumann. "A Data-driven Approach for Noise Reduction in Distantly Supervised Biomedical Relation Extraction." In: *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*. July 2020, pp. 187–194.

[13] Dogu Araci. "Finbert: Financial sentiment analysis with pretrained language models." In: *arXiv preprint arXiv:1908.10063* (2019).

[14] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." In: *arXiv preprint arXiv:1409.0473* (2014).

[15] Riza Theresa Batista-Navarro, Rafal Rak, and Sophia Ananiadou. "Chemistry-specific features and heuristics for developing a CRF-based chemical named entity recogniser." In: *Proceedings of the Fourth BioCreative Challenge Evaluation Workshop*. Vol. 2. 2013, pp. 55–59.

[16] Eyal Beigman and Beata Beigman Klebanov. "Learning with annotation noise." In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. 2009, pp. 280–287.

[17] Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. "Joint entity recognition and relation extraction as a multi-head selection problem." In: *Expert Systems with Applications* 114 (2018), pp. 34–45.

[18] Iz Beltagy, Kyle Lo, and Arman Cohan. "SciBERT: A Pretrained Language Model for Scientific Text." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Nov. 2019, pp. 3615–3620.

[19] Alice M Biggane, Lucy Brading, Philippe Ravaud, Bridget Young, and Paula R Williamson. "Survey indicated that core outcome set development is increasingly including patients, being conducted internationally and using Delphi surveys." In: *Trials* 19.1 (2018), pp. 1–6. ISSN: 1745-6215.

[20] Jari Björne, Suwisa Kaewphan, and Tapio Salakoski. "UTurku: drug named entity recognition and drug-drug interaction extraction using SVM classification and domain knowledge." In: *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. 2013, pp. 651–659.

[21] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. "Enriching word vectors with subword information." In: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 135–146.

[22] Florian Boudin, Jian-Yun Nie, Joan C Bartlett, Roland Grad, Pierre Pluye, and Martin Dawes. "Combining classifiers for robust PICO element detection." In: *BMC medical informatics and decision making* 10.1 (2010), pp. 1–6.

[23] Florian Boudin, Jian-Yun Nie, and Martin Dawes. "Clinical information retrieval using document and PICO structure." In: *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. 2010, pp. 822–830.

[24] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. "A large annotated corpus for learning natural language inference." In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Sept. 2015, pp. 632–642.

[25] Àlex Bravo, Janet Piñero, Núria Queralt-Rosinach, Michael Rautschka, and Laura I Furlong. "Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research." In: *BMC bioinformatics* 16.1 (2015), pp. 1–17.

[26] Austin J Brockmeier, Meizhi Ju, Piotr Przybyła, and Sophia Ananiadou. "Improving reference prioritisation with PICO recognition." In: *BMC medical informatics and decision making* 19.1 (2019), pp. 1–14.

[27] Tom Brown et al. "Language Models are Few-Shot Learners." In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. 2020, pp. 1877–1901.

[28] Brian L Cairns, Rodney D Nielsen, James J Masanz, James H Martin, Martha S Palmer, Wayne H Ward, and Guergana K Savova. "The MiPACQ clinical question answering system." In: *AMIA annual symposium proceedings*. Vol. 2011. American Medical Informatics Association. 2011, p. 171.

[29] Kathi Canese and Sarah Weis. "PubMed: the bibliographic database." In: *The NCBI Handbook* 2 (2013), p. 1.

[30] Samir Chabou and Michal Iglewski. "Combination of conditional random field with a rule based method in the extraction of PICO elements." In: *BMC medical informatics and decision making* 18.1 (2018), pp. 1–14.

[31] Ilias Chalkidis, Manos Fergadiotis, Sotiris Kotitsas, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. "An Empirical Study on Large-Scale Multi-Label Text Classification Including Few and Zero-Shot Labels." In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 7503–7515. DOI: 10.18653/v1/2020.emnlp-main.607. URL: https://aclanthology.org/2020.emnlp-main.607.

[32] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. "LEGAL-BERT: The Muppets straight out of Law School." In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Nov. 2020, pp. 2898–2904.

[33] Yee Seng Chan and Dan Roth. "Exploiting syntactico-semantic structures for relation extraction." In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. 2011, pp. 551–560.

[34] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. "One billion word benchmark for measuring progress in statistical language modeling." In: *arXiv preprint arXiv:1312.3005* (2013).

[35] Daoyuan Chen, Yaliang Li, Kai Lei, and Ying Shen. "Relabel the Noise: Joint Extraction of Entities and Relations via Cooperative Multiagents." In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. July 2020, pp. 5940–5950.

[36] Miao Chen, Ganhui Lan, Fang Du, and Victor Lobanov. "Joint Learning with Pre-trained Transformer on Named Entity Recognition and Relation Extraction Tasks for Clinical Analytics." In: *Proceedings of the 3rd Clinical Natural Language Processing Workshop*. 2020, pp. 234–242.

[37] Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. "Enhanced LSTM for Natural Language Inference." In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. July 2017, pp. 1657–1668.

[38] Mohammad Ali Cheragi, Human Manoocheri, Esmaeil Mohammadnejad, and Syyedeh R Ehsani. "Types and causes of medication errors from nurse's viewpoint." In: *Iranian journal of nursing and midwifery research* 18.3 (2013), p. 228.

[39] Nancy Chinchor and Beth M Sundheim. "MUC-5 evaluation metrics." In: *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland, August 25-27, 1993*. 1993.

[40] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. "Learning phrase representations using RNN encoder-decoder for statistical machine translation." In: *arXiv preprint arXiv:1406.1078* (2014).

[41] Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. "QuAC: Question Answering in Context." In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Oct. 2018, pp. 2174–2184.

[42] Anna Choromanska and Ish Kumar Jain. "Extreme Multiclass Classification Criteria." In: *Computation* 7.1 (2019). ISSN: 2079-3197. DOI: 10.3390/computation7010016. URL: https://www.mdpi.com/2079-3197/7/1/16.

[43] Md Faisal Mahbub Chowdhury and Alberto Lavelli. "FBK-irst: A multi-phase kernel based approach for drug-drug interaction detection and classification that exploits linguistic information." In: *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. 2013, pp. 351–355.

[44] Alexandra Chronopoulou, Christos Baziotis, and Alexandros Potamianos. "An Embarrassingly Simple Approach for Transfer Learning from Pretrained Language Models." 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)*. June 2019, pp. 2089–2095.

[45] Enrico Coiera, Miew Keen Choong, Guy Tsafnat, Peter Hibbert, and William B Runciman. "Linking quality indicators to clinical trials: an automated approach." In: *International Journal*

for Quality in Health Care 29.4 (2017), pp. 571–578. ISSN: 1464-3677.

[46] Ronan Collobert and Jason Weston. "A unified architecture for natural language processing: Deep neural networks with multitask learning." In: *Proceedings of the 25th international conference on Machine learning*. 2008, pp. 160–167.

[47] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. "Supervised Learning of Universal Sentence Representations from Natural Language Inference Data." In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Sept. 2017, pp. 670–680.

[48] Gamal Crichton, Sampo Pyysalo, Billy Chiu, and Anna Korhonen. "A neural network multi-task learning approach to biomedical named entity recognition." In: *BMC bioinformatics* 18.1 (2017), pp. 1–14.

[49] Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. "Template-Based Named Entity Recognition Using BART." In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Aug. 2021, pp. 1835–1845.

[50] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. "Class-balanced loss based on effective number of samples." In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 9268–9277.

[51] Joe Davison, Joshua Feldman, and Alexander M Rush. "Commonsense knowledge mining from pretrained models." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2019, pp. 1173–1178.

[52] Dina Demner-Fushman, Barbara Few, Susan E Hauser, and George Thoma. "Automatically identifying health outcome information in MEDLINE records." In: *Journal of the American Medical Informatics Association* 13.1 (2006), pp. 52–60.

[53] Dina Demner-Fushman and Jimmy Lin. "Answering clinical questions with knowledge-based and statistical techniques." In: *Computational Linguistics* 33.1 (2007), pp. 63–103.

[54] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "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)*. June 2019, pp. 4171–4186.

[55] Susanna Dodd, Mike Clarke, Lorne Becker, Chris Mavergames, Rebecca Fish, and Paula R. Williamson. "A taxonomy has been developed for outcomes in medical research to help improve knowledge discovery." In: *Journal of Clinical Epidemiology* 96 (Apr. 2018), pp. 84–92. ISSN: 18785921. DOI: 10.1016/j.jclinepi.2017.12.020.

[56] Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. "NCBI disease corpus: a resource for disease name recognition and concept normalization." In: *Journal of biomedical informatics* 47 (2014), pp. 1–10.

[57] MM Douglass, GD Cliffford, Andrew Reisner, WJ Long, GB Moody, and RG Mark. "De-identification algorithm for free-text nursing notes." In: *Computers in Cardiology, 2005*. IEEE. 2005, pp. 331–334.

[58] Timothy Dozat and Christopher D Manning. "Deep biaffine attention for neural dependency parsing." In: *arXiv preprint arXiv:1611.01734* (2016).

[59] Cunxiao Du, Zhaozheng Chen, Fuli Feng, Lei Zhu, Tian Gan, and Liqiang Nie. "Explicit interaction model towards text classification." In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 6359–6366.

[60] Kerry Dwan, Carrol Gamble, Paula R Williamson, and Jamie J Kirkham. "Systematic review of the empirical evidence of study publication bias and outcome reporting bias—an updated review." In: *PloS one* 8.7 (2013), e66844.

[61] Sean R Eddy. "What is a hidden Markov model?" In: *Nature biotechnology* 22.10 (2004), pp. 1315–1316.

[62] Katherine Elkins and Jon Chun. "Can GPT-3 pass a writer's Turing Test?" In: *Journal of Cultural Analytics* 5.2 (2020), p. 17212.

[63] Hady Elsahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest, and Elena Simperl. "T-rex: A large scale alignment of natural language with knowledge base triples." In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. 2018.

[64] Angela Fan, Mike Lewis, and Yann Dauphin. "Hierarchical Neural Story Generation." In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. July 2018, pp. 889–898.

[65] Eibe Frank and Remco R Bouckaert. "Naive bayes for text classification with unbalanced classes." In: *European Conference on PKDD*. Springer. 2006, pp. 503–510.

[66] Tianyu Gao, Adam Fisch, and Danqi Chen. "Making Pre-trained Language Models Better Few-shot Learners." In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Aug. 2021, pp. 3816–3830.

[67] Aitor Gonzalez-Agirre, Montserrat Marimon, Ander Intxaurrondo, Obdulia Rabal, Marta Villegas, and Martin Krallinger. "Pharmaconer: Pharmacological substances, compounds and proteins named entity recognition track." In: *Proceedings of The 5th Workshop on BioNLP Open Shared Tasks*. 2019, pp. 1–10.

[68] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." In: *Advances in neural information processing systems* 27 (2014).

[69] Ralph Grishman and Beth M Sundheim. "Message understanding conference-6: A brief history." In: *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*. 1996.

[70] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. "Domain-specific language model pretraining for biomedical natural language processing." In: *ACM Transactions on Computing for Healthcare (HEALTH)* 3.1 (2021), pp. 1–23.

[71] R Guillen et al. "Automated de-identification and categorization of medical records." In: *i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data*. Vol. 116. 2006.

[72] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. "Don't Stop Pretraining: Adapt Language Models to Domains and Tasks." In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 8342–8360. DOI: 10.18653/v1/2020.acl-main.740. URL: https://aclanthology.org/2020.acl-main.740.

[73] Marie J Hansen, Nana Rasmussen, and Grace Chung. "A method of extracting the number of trial participants from abstracts describing randomized controlled trials." In: *Journal of Telemedicine and Telecare* 14.7 (2008), pp. 354–358.

[74] Yaru Hao, Li Dong, Furu Wei, and Ke Xu. "Visualizing and Understanding the Effectiveness of BERT." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Nov. 2019, pp. 4143–4152.

[75] Benjamin Heinzerling and Kentaro Inui. "Language Models as Knowledge Bases: On Entity Representations, Storage Capacity, and Paraphrased Queries." In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Apr. 2021, pp. 1772–1791.

[76] María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. "The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions." In: *Journal of biomedical informatics* 46.5 (2013), pp. 914–920.

[77] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory." In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[78] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. "Training Compute-Optimal Large Language Models." In: *arXiv preprint arXiv:2203.15556* (2022).

[79] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. "The curious case of neural text degeneration." In: *arXiv preprint arXiv:1904.09751* (2019).

[80] Jeremy Howard and Sebastian Ruder. "Universal Language Model Fine-tuning for Text Classification." In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. July 2018, pp. 328–339.

[81] Linmei Hu, Luhao Zhang, Chuan Shi, Liqiang Nie, Weili Guan, and Cheng Yang. "Improving Distantly-Supervised Relation Extraction with Joint Label Embedding." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Nov. 2019, pp. 3821–3829.

[82] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. "Learning deep representation for imbalanced classification." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 5375–5384.

[83] Chung-Chi Huang and Zhiyong Lu. "Community challenges in biomedical text mining over 10 years: success, failure and the future." In: *Briefings in bioinformatics* 17.1 (2016), pp. 132–144.

[84] Ke-Chun Huang, I-Jen Chiang, Furen Xiao, Chun-Chih Liao, Charles Chih-Ho Liu, and Jau-Min Wong. "PICO element detection in medical text without metadata: Are first sentences enough?" In: *Journal of biomedical informatics* 46.5 (2013), pp. 940–946.

[85] Ke-Chun Huang, Charles Chih-Ho Liu, Shung-Shiang Yang, Furen Xiao, Jau-Min Wong, Chun-Chih Liao, and I-Jen Chiang. "Classification of PICO elements by text features systematically extracted from PubMed abstracts." In: *2011 IEEE International Conference on Granular Computing*. IEEE. 2011, pp. 279–283.

[86] Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. "Clinicalbert: Modeling clinical notes and predicting hospital readmission." In: *arXiv preprint arXiv:1904.05342* (2019).

[87] Weipeng Huang, Xingyi Cheng, Taifeng Wang, and Wei Chu. "BERT-based multi-head selection for joint entity-relation extraction." In: *CCF international conference on natural language processing and chinese computing*. Springer. 2019, pp. 713–723.

[88] Xiaoli Huang, Jimmy Lin, and Dina Demner-Fushman. "Evaluation of PICO as a knowledge representation for clinical questions." In: 2006 (2006), p. 359.

[89] Comet Initiative. *Comet Initiative*. Accessed Feb. 15, 2022. Core Outcome Measures in Effectiveness Trials (COMET). URL: https://www.comet-initiative.org/.

[90] Ander Intxaurrondo, Mihai Surdeanu, Oier Lopez De Lacalle, and Eneko Agirre. "Removing noisy mentions for distant supervision." In: *Procesamiento del lenguaje natural* 51 (2013), pp. 41–48.

[91] Zongcheng Ji, Tian Xia, Mei Han, and Jing Xiao. "A Neural Transition-based Joint Model for Disease Named Entity Recognition and Normalization." In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2021, pp. 2819–2827.

[92] Wei Jia, Dai Dai, Xinyan Xiao, and Hua Wu. "ARNOR: Attention regularization based noise reduction for distant supervision relation classification." In: *Proceedings of the 57th annual meeting of the association for computational linguistics*. 2019, pp. 1399–1408.

[93] Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. "X-FACTR: Multilingual Factual Knowledge Retrieval from Pretrained Language Models." In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Nov. 2020, pp. 5943–5959.

[94] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. "How can we know what language models know?" In: *Transactions of the Association for Computational Linguistics* 8 (2020), pp. 423–438.

[95] Di Jin and Peter Szolovits. "Advancing PICO Element Detection in Medical Text via Deep Neural Networks." In: *CoRR* (2018).

[96] Di Jin and Peter Szolovits. "Pico element detection in medical text via long short-term memory neural networks." In: *Proceedings of the BioNLP 2018 workshop*. 2018, pp. 67–75.

[97] Qiao Jin, Bhuwan Dhingra, William Cohen, and Xinghua Lu. "Probing Biomedical Embeddings from Language Models." In: *Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*. June 2019, pp. 82–89.

[98] Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. "PubMedQA: A Dataset for Biomedical Research Question Answering." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Nov. 2019, pp. 2567–2577.

[99] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-Wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. "MIMIC-III, a freely accessible critical care database." In: *Scientific data* 3.1 (2016), pp. 1–9.

[100] Siddhartha R Jonnalagadda, Pawan Goyal, and Mark D Huffman. "Automating data extraction in systematic reviews: a systematic review." In: *Systematic reviews* 4.1 (2015), pp. 1–16.

[101] Tian Kang, Shirui Zou, and Chunhua Weng. "Pretraining to Recognize PICO Elements from Randomized Controlled Trial Literature." In: *Studies in health technology and informatics* 264 (2019), p. 188.

[102] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. "Scaling laws for neural language models." In: *arXiv preprint arXiv:2001.08361* (2020).

[103] Akbar Karimi, Leonardo Rossi, and Andrea Prati. "Adversarial Training for Aspect-Based Sentiment Analysis with BERT." In: *2020 25th International Conference on Pattern Recognition (ICPR)*. 2021, pp. 8797–8803. DOI: 10.1109/ICPR48806.2021.9412167.

[104] Arzoo Katiyar and Claire Cardie. "Going out on a limb: Joint extraction of entity mentions and relations without dependency trees." In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2017, pp. 917–928.

[105] Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. "UNIFIEDQA: Crossing Format Boundaries with a Single QA System." In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Nov. 2020, pp. 1896–1907.

[106] Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Nigel Collier. "Introduction to the bio-entity recognition task at JNLPBA." In: *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*. Citeseer. 2004, pp. 70–75.

[107] Su Nam Kim, David Martinez, Lawrence Cavedon, and Lars Yencken. "Automatic classification of sentences to support evidence based medicine." In: *BMC bioinformatics*. Vol. 12. 2. BioMed Central. 2011, pp. 1–10.

[108] Sun Kim, Haibin Liu, Lana Yeganova, and W John Wilbur. "Extracting drug–drug interactions from literature using a rich feature-based linear kernel approach." In: *Journal of biomedical informatics* 55 (2015), pp. 23–30.

[109] Yoon Kim. "Convolutional Neural Networks for Sentence Classification." In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Oct. 2014, pp. 1746–1751.

[110] Svetlana Kiritchenko, Berry De Bruijn, Simona Carini, Joel Martin, and Ida Sim. "ExaCT: automatic extraction of clinical trial characteristics from journal publications." In: *BMC medical informatics and decision making* 10.1 (2010), pp. 1–17.

[111] Jamie J Kirkham, Kerry M Dwan, Douglas G Altman, Carrol Gamble, Susanna Dodd, Rebecca Smyth, and Paula R Williamson. "The impact of outcome reporting bias in randomised controlled trials on a cohort of systematic reviews." In: *Bmj* 340 (2010).

[112] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. "Moses: Open source toolkit for statistical machine translation." In: *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*. 2007, pp. 177–180.

[113] Anna Koroleva, Sanjay Kamath, and Patrick Paroubek. "Measuring semantic similarity of clinical trial outcomes using deep pre-trained language representations." In: *Journal of Biomedical Informatics: X* 4 (2019), p. 100058.

[114] Anna Koroleva and Patrick Paroubek. "Extracting relations between outcomes and significance levels in Randomized Controlled Trials (RCTs) publications." In: *Proceedings of the 18th BioNLP Workshop and Shared Task*. 2019, pp. 359–369.

[115] Jens Kringelum, Sonny Kim Kjaerulff, Søren Brunak, Ole Lund, Tudor I Oprea, and Olivier Taboureau. "ChemProt-3.0: a global chemical biology diseases mapping." In: *Database* 2016 (2016).

[116] John Lafferty, Andrew McCallum, and Fernando CN Pereira. "Conditional random fields: Probabilistic models for segmenting and labeling sequence data." In: *Proceedings of the 18th International Conference on Machine Learning (ICML 2001)*. 2001, pp. 282–289.

[117] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. "Neural Architectures for Named Entity Recognition." In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. June 2016, pp. 260–270.

[118] Tamar Lavee, Lili Kotlerman, Matan Orbach, Yonatan Bilu, Michal Jacovi, Ranit Aharonov, and Noam Slonim. "Crowdsourcing annotation of complex NLU tasks: A case study of argumentative content annotation." In: *Proceedings of the First Workshop on Aggregating and Analysing Crowdsourced Annotations for NLP*. 2019, pp. 29–38.

[119] Phong Le and Ivan Titov. "Distant Learning for Entity Linking with Automatic Noise Detection." In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. July 2019, pp. 4081–4090.

[120] Robert Leaman and Zhiyong Lu. "TaggerOne: joint named entity recognition and normalization with semi-Markov Models." In: *Bioinformatics* 32.18 (2016), pp. 2839–2846.

[121] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. "BioBERT: a pre-trained biomedical language representation model for biomedical text mining." In: *Bioinformatics* 36.4 (2020), pp. 1234–1240.

[122] Eric Lehman, Jay DeYoung, Regina Barzilay, and Byron C. Wallace. "Inferring Which Medical Treatments Work from Reports of Clinical Trials." 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)*. June 2019, pp. 3705–3717.

[123] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension." In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. July 2020, pp. 7871–7880.

[124] Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. "Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art." In: *Proceedings of the 3rd Clinical Natural Language Processing Workshop*. 2020, pp. 146–157.

[125] Fei Li, Meishan Zhang, Guohong Fu, and Donghong Ji. "A neural joint model for entity and relation extraction from biomedical text." In: *BMC bioinformatics* 18.1 (2017), pp. 1–11.

[126] Gang Li, Cathy Wu, and K Vijay-Shanker. "Noise reduction methods for distantly supervised biomedical relation extraction." In: *BioNLP 2017*. 2017, pp. 184–193.

[127] Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. "BioCreative V CDR task corpus: a resource for chemical disease relation extraction." In: *Database* 2016 (2016).

[128] Xiang Lisa Li and Percy Liang. "Prefix-Tuning: Optimizing Continuous Prompts for Generation." In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Aug. 2021, pp. 4582–4597.

[129] Sangrak Lim and Jaewoo Kang. "Chemical–gene relation extraction using recursive neural network." In: *Database* 2018 (2018).

[130] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. "Focal loss for dense object detection." In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2980–2988.

[131] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. "A structured self-attentive sentence embedding." In: *arXiv preprint arXiv:1703.03130* (2017).

[132] Marianne Lisby, Lars Peter Nielsen, and Jan Mainz. "Errors in the medication process: frequency, type, and potential clinical consequences." In: *International journal for quality in health care* 17.1 (2005), pp. 15–22.

[133] Liqun Liu, Funan Mu, Pengyu Li, Xin Mu, Jing Tang, Xingsheng Ai, Ran Fu, Lifeng Wang, and Xing Zhou. "Neuralclassifier: An open-source neural hierarchical multi-label text classification toolkit." In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. 2019, pp. 87–92.

[134] Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. "Linguistic Knowledge and Transferability of Contextual Representations." 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)*. June 2019, pp. 1073–1094.

[135] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing." In: *arXiv preprint arXiv:2107.13586* (2021).

[136] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. "Roberta: A robustly optimized bert pretraining approach." In: *arXiv preprint arXiv:1907.11692* (2019).

[137] Yinxia Lou, Yue Zhang, Tao Qian, Fei Li, Shufeng Xiong, and Donghong Ji. "A transition-based joint model for disease named entity recognition and normalization." In: *Bioinformatics* 33.15 (2017), pp. 2363–2371.

[138] Daniel Loureiro and Alípio Jorge. "Language Modelling Makes Sense: Propagating Representations through WordNet for Full-Coverage Word Sense Disambiguation." In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. July 2019, pp. 5682–5691.

[139] Jesús Lovón-Melgarejo, Laure Soulier, Karen Pinel-Sauvagnat, and Lynda Tamine. "Studying catastrophic forgetting in neural ranking models." In: *European Conference on Information Retrieval*. Springer. 2021, pp. 375–390.

[140] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. "Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity." In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. May 2022, pp. 8086–8098.

[141] Bingfeng Luo, Yansong Feng, Zheng Wang, Zhanxing Zhu, Songfang Huang, Rui Yan, and Dongyan Zhao. "Learning with Noise: Enhance Distantly Supervised Relation Extraction with Dynamic Transition Matrix." In: *Proceedings of the 55th Annual*

Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). July 2017, pp. 430–439.

[142] Hongyin Luo, Shang-Wen Li, and James Glass. "Knowledge Grounded Conversational Symptom Detection with Graph Memory Networks." In: *Proceedings of the 3rd Clinical Natural Language Processing Workshop*. Nov. 2020, pp. 136–145.

[143] Mingbo Ma, Kai Zhao, Liang Huang, Bing Xiang, and Bowen Zhou. "Jointly trained sequential labeling and classification by sparse attention neural networks." In: *arXiv preprint arXiv:1709.10191* (2017).

[144] Benjamin Markines, Ciro Cattuto, and Filippo Menczer. "Social spam detection." In: *Proceedings of the 5th international workshop on adversarial information retrieval on the web*. 2009, pp. 41–48.

[145] Aurelie Mascio, Zeljko Kraljevic, Daniel Bean, Richard Dobson, Robert Stewart, Rebecca Bendayan, and Angus Roberts. "Comparative Analysis of Text Classification Approaches in Electronic Health Records." In: *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*. July 2020, pp. 86–94.

[146] Sherri Matis-Mitchell, Phoebe Roberts, Catalina O Tudor, and Cecilia N Arighi. "BioCreative IV interactive task." In: *Fourth BioCreative Challenge Evaluation Workshop. Bethesda, MD*. 2013, pp. 190–203.

[147] Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. "The Natural Language Decathlon: Multitask Learning as Question Answering." In: *arXiv preprint arXiv:1806.08730* (2018).

[148] George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alexander Wong. "UmlsBERT: Clinical Domain Knowledge Augmentation of Contextual Embeddings Using the Unified Medical Language System Metathesaurus." In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. June 2021, pp. 1744–1753.

[149] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. "Distributed representations of words and phrases and their compositionality." In: *Advances in neural information processing systems* 26 (2013).

[150] George A Miller. "WordNet: a lexical database for English." In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[151] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. "Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?" In: *arXiv e-prints* (Feb. 2022).

[152] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. "Distant supervision for relation extraction without labeled data." In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. 2009, pp. 1003–1011.

[153] Makoto Miwa and Yutaka Sasaki. "Modeling joint entity and relation extraction with table representation." In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014, pp. 1858–1869.

[154] Usman Naseem, Matloob Khushi, Vinay Reddy, Sakthivel Rajendran, Imran Razzak, and Jinman Kim. "Bioalbert: A simple and effective pre-trained language model for biomedical named entity recognition." In: *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2021, pp. 1–7.

[155] National Center for Biotechnology Information (NCBI). *Medline*. Accessed Feb. 10, 2022. National Library of Medicine (NLM). URL: https://www.nlm.nih.gov/medline/medline_overview.html.

[156] National Institute of Health (NIH). Accessed Feb. 21 2022. National Library of Medicine (NLM). URL: https://geneticassociationdb.nih.gov/.

[157] Ishna Neamatullah, Margaret M Douglass, H Lehman Li-wei, Andrew Reisner, Mauricio Villarroel, William J Long, Peter Szolovits, George B Moody, Roger G Mark, and Gari D Clifford. "Automated de-identification of free-text medical records." In: *BMC medical informatics and decision making* 8.1 (2008), pp. 1–17.

[158] Farhad Nooralahzadeh, Jan Tore Lonning, and Lilja Ovrelid. "Reinforcement-based denoising of distantly supervised NER with partial annotation." In: *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 225–233. DOI: 10.18653/v1/D19-6125. URL: https://aclanthology.org/D19-6125.

[159] Dcana E Northup, Margaret Moore-West, Betty Skipper, and Sallie R Teaf. "Characteristics of clinical information-searching: investigation using critical incident technique." In: *Journal of Medical Education* 58.11 (1983), pp. 873–881.

[160] Joel Nothman, Hanmin Qin, and Roman Yurchak. "Stop word lists in free open-source software packages." In: *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*. 2018, pp. 7–12.

[161] Benjamin Nye, Junyi Jessy Li, Roma Patel, Yinfei Yang, Iain J. Marshall, Ani Nenkova, and Byron C. Wallace. "A Corpus with Multi-Level Annotations of Patients, Interventions and Outcomes to Support Language Processing for Medical Literature." In: *Proc. of the 56th Annual Meeting of the Association for Computational Linguistics*. 2018, pp. 197–207.

[162] Tomoko Ohta, Yuka Tateisi, Jin-Dong Kim, Hideki Mima, and Junichi Tsujii. "The GENIA corpus: An annotated research abstract corpus in molecular biology domain." In: *Proceedings of the human language technology conference*. Citeseer. 2002, pp. 73–77.

[163] Arzucan Özgür, Thuy Vu, Güneş Erkan, and Dragomir R Radev. "Identifying gene-disease associations using centrality on a literature mined gene-interaction network." In: *Bioinformatics* 24.13 (2008), pp. i277–i285.

[164] Alexander Pak and Patrick Paroubek. "Twitter as a corpus for sentiment analysis and opinion mining." In: *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. 2010.

[165] Anusri Pampari, Preethi Raghavan, Jennifer Liang, and Jian Peng. "emrQA: A Large Corpus for Question Answering on Electronic Medical Records." In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Oct. 2018, pp. 2357–2368.

[166] Jeffrey Pennington, Richard Socher, and Christopher D Manning. "Glove: Global vectors for word representation." In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, pp. 1532–1543.

[167] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. "Deep Contextualized Word Representations." In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. June 2018, pp. 2227–2237.

[168] Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. "Knowledge Enhanced Contextual Word Representations." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Nov. 2019, pp. 43–54.

[169] Matthew E. Peters, Sebastian Ruder, and Noah A. Smith. "To Tune or Not to Tune? Adapting Pretrained Representations to Diverse Tasks." In: *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*. Aug. 2019, pp. 7–14.

[170] Fabio Petroni, Tim Rocktaschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. "Language Models as Knowledge Bases?" In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 2463–2473. DOI: 10.18653/v1/D19-1250. URL: https://aclanthology.org/D19-1250.

[171] John Platt et al. "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods." In: *Advances in large margin classifiers* 10.3 (1999), pp. 61–74.

[172] Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. "Semeval-2016 task 5: Aspect based sentiment analysis." In: *International workshop on semantic evaluation*. 2016, pp. 19–30.

[173] Guanghui Qin and Jason Eisner. "Learning How to Ask: Querying LMs with Mixtures of Soft Prompts." In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. June 2021, pp. 5203–5212.

[174] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. "Exploring the limits of transfer learning with a unified text-to-text transformer." In: *arXiv preprint arXiv:1910.10683* (2019).

[175] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. "SQuAD: 100,000+ Questions for Machine Comprehension of Text." In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Nov. 2016, pp. 2383–2392.

[176] Vikas Raunak, Vivek Gupta, and Florian Metze. "Effective dimensionality reduction for word embeddings." In: *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*. 2019, pp. 235–243.

[177] Radim Rehurek and Petr Sojka. "Software framework for topic modelling with large corpora." In: *In Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks*. Citeseer. 2010.

[178] Kevin Reschke, Martin Jankowiak, Mihai Surdeanu, Christopher D Manning, and Dan Jurafsky. "Event extraction using distant supervision." In: *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. 2014, pp. 4527–4531.

[179] Alexey Romanov and Chaitanya Shivade. "Lessons from Natural Language Inference in the Clinical Domain." In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Oct. 2018, pp. 1586–1596.

[180] Sebastian Ruder. "An overview of multi-task learning in deep neural networks." In: *arXiv preprint arXiv:1706.05098* (2017).

[181] David L Sackett, William M C Rosenberg, J A Muir Gray, R Brian Haynes, and W Scott Richardson. "Evidence based medicine: what it is and what it isn't." In: *BMJ* 312.7023 (1996), pp. 71–72. ISSN: 0959-8138. DOI: 10.1136/bmj.312.7023.71. eprint: https://www.bmj.com/content. URL: https://www.bmj.com/content/312/7023/71.

[182] Erik F Sang and Jorn Veenstra. "Representing text chunks." In: *Proceedings of the European Chapter of the Association for Computational Linguistics (EACL)*. 1999.

[183] Beatrice Santorini. "Part-of-speech tagging guidelines for the Penn Treebank Project." In: (1990).

[184] Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. "Social IQa: Commonsense Reasoning about Social Interactions." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 4463–4473. DOI: 10.18653/v1/D19-1454. URL: https://aclanthology.org/D19-1454.

[185] Eric Sayers. "The E-utilities in-depth: parameters, syntax and more." In: *Entrez Programming Utilities Help [Internet]* (2009).

[186] Timo Schick and Hinrich Schütze. "Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference." In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Apr. 2021, pp. 255–269.

[187] Timo Schick and Hinrich Schütze. "It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners." In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. June 2021, pp. 2339–2352.

[188] Isabel Segura-Bedmar, Paloma Martínez, and César de Pablo-Sánchez. "Extracting drug-drug interactions from biomedical texts." In: *BMC bioinformatics* 11.5 (2010), pp. 1–2.

[189] Shreyas Sharma and Ron Daniel Jr. "BioFLAIR: Pretrained pooled contextualized embeddings for biomedical sequence labeling tasks." In: *arXiv preprint arXiv:1908.05760* (2019).

[190] Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. "AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts." In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Nov. 2020, pp. 4222–4235.

[191] Borbála Siklósi and Attila Novák. "Restoring the intended structure of Hungarian ophthalmology documents." In: *Proceedings of BioNLP 15*. 2015, pp. 152–157.

[192] L Smith, Thomas Rindflesch, and W John Wilbur. "MedPost: a part-of-speech tagger for bioMedical text." In: *Bioinformatics* 20.14 (2004), pp. 2320–2321.

[193] Larry Smith, Lorraine K Tanabe, Rie Johnson nee Ando, Cheng-Ju Kuo, I-Fang Chung, Chun-Nan Hsu, Yu-Shi Lin, Roman Klinger, Christoph M Friedrich, Kuzman Ganchev, et al. "Overview of BioCreative II gene mention recognition." In: *Genome biology* 9.2 (2008), pp. 1–19.

[194] Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, et al. "Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model." In: *arXiv preprint arXiv:2201.11990* (2022).

[195] Mark Starr, Iain Chalmers, Mike Clarke, and Andrew D Oxman. "The origins, evolution, and future of The Cochrane Database of Systematic Reviews." In: *International journal of technology assessment in health care* 25.S1 (2009), pp. 182–195.

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

[197] Nikolaos Stylianou, Gerasimos Razis, Dimitrios G Goulis, and Ioannis Vlahavas. "EBM+: Advancing Evidence-Based Medicine via two level automatic identification of Populations, Interventions, Outcomes in medical literature." In: *Artificial Intelligence in Medicine* 108 (2020), p. 101949.

[198] Alane Suhr, Clara Vania, Nikita Nangia, Maarten Sap, Mark Yatskar, Samuel Bowman, and Yoav Artzi. "Crowdsourcing Beyond Annotation: Case Studies in Benchmark Data Collection." In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*. 2021, pp. 1–6.

[199] Cong Sun and Zhihao Yang. "Transfer learning in biomedical named entity recognition: An evaluation of BERT in the PharmaCoNER task." In: *Proceedings of The 5th Workshop on BioNLP Open Shared Tasks*. 2019, pp. 100–104.

[200] Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. "Can Language Models be Biomedical Knowledge Bases?" In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Nov. 2021, pp. 4723–4734.

[201] Mihai Surdeanu. "Overview of the TAC2013 Knowledge Base Population Evaluation: English Slot Filling and Temporal Slot Filling." In: *TAC* 8 (2013), p. 2.

[202] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. "Intriguing properties of neural networks." English (US). In: 2nd International Conference on Learning Representations, ICLR 2014 ; Conference date: 14-04-2014 Through 16-04-2014. Jan. 2014.

[203] Kai Sheng Tai, Richard Socher, and Christopher D. Manning. "Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks." 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)*. July 2015, pp. 1556–1566.

[204] Shingo Takamatsu, Issei Sato, and Hiroshi Nakagawa. "Reducing wrong labels in distant supervision for relation extraction." In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2012, pp. 721–729.

[205] Lorraine Tanabe and W John Wilbur. "Tagging gene and protein names in biomedical text." In: *Bioinformatics* 18.8 (2002), pp. 1124–1132.

[206] Toomas Timpka, Marie Ekström, and Per Bjurulf. "Information needs and information seeking behaviour in primary health care." In: *Scandinavian journal of primary health care* 7.2 (1989), pp. 105–109.

[207] Erik F. Tjong Kim Sang and Fien De Meulder. "Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition." In: *Proceedings of the Seventh Conference on*

Natural Language Learning at HLT-NAACL 2003. 2003, pp. 142–147.

[208] George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. "An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition." In: *BMC bioinformatics* 16.1 (2015), pp. 1–28.

[209] Yoshimasa Tsuruoka, Yuka Tateishi, Jin-Dong Kim, Tomoko Ohta, John McNaught, Sophia Ananiadou, and Jun'ichi Tsujii. "Developing a robust part-of-speech tagger for biomedical text." In: *Panhellenic Conference on Informatics*. Springer. 2005, pp. 382–392.

[210] Peter Tugwell, Maarten Boers, Peter Brooks, Lee Simon, Vibeke Strand, and Leanne Idzerda. "OMERACT: an international initiative to improve outcome measurement in rheumatology." In: *Trials* 8.1 (2007), pp. 1–6.

[211] Özlem Uzuner, Yuan Luo, and Peter Szolovits. "Evaluating the state-of-the-art in automatic de-identification." In: *Journal of the American Medical Informatics Association* 14.5 (2007), pp. 550–563. ISSN: 1527-974X.

[212] Erik M Van Mulligen, Annie Fourrier-Reglat, David Gurwitz, Mariam Molokhia, Ainhoa Nieto, Gianluca Trifiro, Jan A Kors, and Laura I Furlong. "The EU-ADR corpus: annotated drugs, diseases, targets, and their relationships." In: *Journal of biomedical informatics* 45.5 (2012), pp. 879–884.

[213] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." In: *Advances in neural information processing systems* 30 (2017).

[214] Byron C Wallace, Joël Kuiper, Aakash Sharma, Mingxi Zhu, and Iain J Marshall. "Extracting PICO sentences from clinical trial reports using supervised distant supervision." In: *The Journal of Machine Learning Research* 17.1 (2016), pp. 4572–4596. ISSN: 1532-4435.

[215] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding." In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Nov. 2018, pp. 353–355.

[216] Benyou Wang, Lifeng Shang, Christina Lioma, Xin Jiang, Hao Yang, Qun Liu, and Jakob Grue Simonsen. "On position embeddings in bert." In: *International Conference on Learning Representations*. 2020.

[217] Yanshan Wang, Naveed Afzal, Sijia Liu, Majid Rastegar-Mojarad, Liwei Wang, Feichen Shen, Sunyang Fu, and Hongfang Liu. "Overview of the BioCreative/OHNLP challenge 2018 task 2: clinical semantic textual similarity." In: *Proceedings of the BioCreative/OHNLP Challenge* 2018 (2018).

[218] Yanshan Wang, Liwei Wang, Majid Rastegar-Mojarad, Sungrim Moon, Feichen Shen, Naveed Afzal, Sijia Liu, Yuqun Zeng, Saeed Mehrabi, Sunghwan Sohn, et al. "Clinical information extraction applications: a literature review." In: *Journal of biomedical informatics* 77 (2018), pp. 34–49.

[219] Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. "Assessing the state of the art in biomedical relation extraction: overview of the BioCreative V chemical-disease relation (CDR) task." In: *Database* 2016 (2016).

[220] Wei Wei, Zanbo Wang, Xianling Mao, Guangyou Zhou, Pan Zhou, and Sheng Jiang. "Position-aware self-attention based neural sequence labeling." In: *Pattern Recognition* 110 (2021), p. 107636.

[221] Zhongyu Wei, Qianlong Liu, Baolin Peng, Huaixiao Tou, Ting Chen, Xuan-Jing Huang, Kam-Fai Wong, and Xiang Dai. "Task-oriented dialogue system for automatic diagnosis." In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2018, pp. 201–207.

[222] Ben Wellner. "Weakly supervised learning methods for improving the quality of gene name normalization data." In: *Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*. 2005, pp. 1–8.

[223] Georg Wiese, Dirk Weissenborn, and Mariana Neves. "Neural Domain Adaptation for Biomedical Question Answering." In: *CoNLL 2017* (2017), p. 281.

[224] Adina Williams, Nikita Nangia, and Samuel Bowman. "A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference." In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. June 2018, pp. 1112–1122.

[225] Paula R Williamson, Douglas G Altman, Heather Bagley, Karen L Barnes, Jane M Blazeby, Sara T Brookes, Mike Clarke, Elizabeth Gargon, Sarah Gorst, Nicola Harman, et al. *The COMET handbook: version 1.0*. 2017.

[226] Paula R Williamson, Douglas G Altman, Jane M Blazeby, Mike Clarke, Declan Devane, Elizabeth Gargon, and Peter Tugwell. "Developing core outcome sets for clinical trials: issues to consider." In: *Trials* 13.1 (2012), pp. 1–8.

[227] Wishart, DS and Feunang, YD and Guo, AC and Lo, EJ and Marcu, A and Grant, JR and Sajed, T and Johnson, D and Li, C and Sayeeda, Z and Assempour, N and Iynkkaran, I and Liu, Y and Maciejewski, A and Gale, N and Wilson, A and Chin, L and Cummings, R and Le, D and Pon, A and Knox, C and Wilson M. Accessed Feb. 10, 2022. Nucleic Acids Research. URL: https://go.drugbank.com/.

[228] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. "Google's neural machine translation system: Bridging the gap between human and machine translation." In: *arXiv preprint arXiv:1609.08144* (2016).

[229] Lin Xiao, Xin Huang, Boli Chen, and Liping Jing. "Label-specific document representation for multi-label text classification." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2019, pp. 466–475.

[230] Hu Xu, Bing Liu, Lei Shu, and Philip Yu. "BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis." In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 2324–2335. DOI: 10.18653/v1/N19-1242. URL: https://aclanthology.org/N19-1242.

[231] Puyang Xu and Ruhi Sarikaya. "Convolutional neural network based triangular crf for joint intent detection and slot filling." In: *2013 ieee workshop on automatic speech recognition and understanding*. IEEE. 2013, pp. 78–83.

[232] Wei Xu, Raphael Hoffmann, Le Zhao, and Ralph Grishman. "Filling knowledge base gaps for distant supervision of relation extraction." In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2013, pp. 665–670.

[233] Yaosheng Yang, Wenliang Chen, Zhenghua Li, Zhengqiu He, and Min Zhang. "Distantly supervised NER with partial annotation learning and reinforcement learning." In: *Proceedings of the 27th International Conference on Computational Linguistics*. 2018, pp. 2159–2169.

[234] Yiming Yang and Jan O Pedersen. "A comparative study on feature selection in text categorization." In: *Icml*. Vol. 97. 412-420. Nashville, TN, USA. 1997, p. 35.

[235] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. "XLNet: Generalized Autoregressive Pretraining for Language Understanding." In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper/2019/file/dc6a7e655d7e5840e66733e9ee67cc69-Paper.pdf.

[236] Yunzhi Yao, Shaohan Huang, Wenhui Wang, Li Dong, and Furu Wei. "Adapt-and-Distill: Developing Small, Fast and Effective Pretrained Language Models for Domains." In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Aug. 2021, pp. 460–470.

[237] Qinyuan Ye, Liyuan Liu, Maosen Zhang, and Xiang Ren. "Looking Beyond Label Noise: Shifted Label Distribution Matters in Distantly Supervised Relation Extraction." In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Nov. 2019, pp. 3841–3850.

[238] Alexander Yeh, Alexander Morgan, Marc Colosimo, and Lynette Hirschman. "BioCreAtIvE task 1A: gene mention finding evaluation." In: *BMC bioinformatics* 6.1 (2005), pp. 1–10.

[239] Wonjin Yoon, Richard Jackson, Aron Lagerberg, and Jaewoo Kang. "Sequence tagging for biomedical extractive question answering." In: *Bioinformatics* 38.15 (2022), pp. 3794–3801.

[240] Wenhao Yu, Chenguang Zhu, Zaitang Li, Zhiting Hu, Qingyun Wang, Heng Ji, and Meng Jiang. "A survey of knowledge-enhanced text generation." In: *arXiv preprint arXiv:2010.04389* (2020).

[241] Zheng Yuan, Yijia Liu, Chuanqi Tan, Songfang Huang, and Fei Huang. "Improving Biomedical Pretrained Language Models with Knowledge." In: *Proceedings of the 20th Workshop on Biomedical Language Processing, BioNLP@NAACL-HLT 2021, Online, June 11, 2021*. Association for Computational Linguistics, 2021, pp. 180–190.

[242] Zheng Yuan, Yijia Liu, Chuanqi Tan, Songfang Huang, and Fei Huang. "Improving Biomedical Pretrained Language Models with Knowledge." In: *Proceedings of the 20th Workshop on Biomedical Language Processing, BioNLP@NAACL-HLT 2021, Online, June 11, 2021*. Association for Computational Linguistics, 2021, pp. 180–190.

[243] Zhang Yun-tao, Gong Ling, and Wang Yong-cheng. "An improved TF-IDF approach for text classification." In: *Journal of Zhejiang University-Science A* 6.1 (2005), pp. 49–55.

[244] Shu Zhang, Dequan Zheng, Xinchen Hu, and Ming Yang. "Bidirectional long short-term memory networks for relation classification." In: *Proc of the 29th PACLIC'2015*. 2015, pp. 73–78.

[245] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. "Position-aware attention and supervised data improve slot filling." In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2017, pp. 35–45.

[246] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. "ERNIE: Enhanced Language Representation with Informative Entities." In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. July 2019, pp. 1441–1451.

[247] Sendong Zhao, Ting Liu, Sicheng Zhao, and Fei Wang. "A neural multi-task learning framework to jointly model medical named entity recognition and normalization." In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 817–824.

[248] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. "Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books." In: *The IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.