Adverse Drug Event Information Extraction from Medical Narratives

using Ensemble Learning & Deep Learning

by
Susmitha Wunnava

A Dissertation
Submitted to the Faculty
of the
WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the
Degree of Doctor of Philosophy
in
Data Science

by

November 18, 2020

APPROVED:

Professor Elke A. Rundensteiner Worcester Polytechnic Institute Advisor	Professor Xiangnan Kong Worcester Polytechnic Institute Committee Member
Professor Eleanor T. Loiacono Worcester Polytechnic Institute Committee Member	Dr. Marni Hall Vice President Clinical Evidence, Head US Regulatory Science & Strategy. IQVIA Committee Member

Abstract

An adverse drug event (ADE) is an injury resulting from medical intervention related to a drug. Many ADEs are detected only during the post-marketing phase of the drug when it is used by a more diverse population than during clinical trials. Early detection of the ADE incidents is crucial for timely assessment, mitigation and prevention of future occurrences of ADEs. Natural Language Processing (NLP) techniques towards ADE information detection from medical narratives provides an effective way of post-marketing drug safety monitoring and pharmacovigilance.

My dissertation studies the problem of detecting ADE information from medical narratives at different levels of granularity: word-level, sentence-level and multi-grained (word-level + sentence-level) using supervised machine learning techniques.

In this dissertation research, we first propose an Ensemble learning approach for fine-grained word-level information detection. Existing supervised machine learning approaches towards biomedical Named Entity Recognition (NER) are limited in their ability to identify certain entity types and result in significant performance difference in terms of accuracy. Another critical problem faced by NER in the biomedical context is that the data is highly skewed for these challenging entity types. We propose a novel methodology called Tiered Ensemble Learning System with Diversity (TELS-D) to address the above challenges in NER. We propose a balanced, under-sampled bagging strategy that is dependent on the level of imbalance to overcome the class imbalance problem. Next we propose an ensemble of heterogeneous recognizers approach that leverages a novel ensemble combiner.

Second, we propose Sequence labeling for word-level information detection using deep learning. Although Electronic health records (EHR) contain valuable ADE information, the EHR text tends to be noisy and comprised of medical and non-medical abbreviations, acronyms, numbers, misspelled words and semantic type ambiguity among certain named entities - making it difficult to detect critical information. We propose the Dual-Level Embedding for Adverse Drug Event Detection framework (DLADE) by adapting a three-layered, deep learning RNN architecture of (1) Bi-directional Long Short-Term Memory (Bi-LSTM) for character-level word representation to encode the morphological features of the medical terminology, (2) Bi-LSTM for capturing

the contextual information of each word within a sentence, and (3) Conditional Random Fields for the final label prediction by also considering the surrounding words. In addition, we propose a rule-based EHR text preprocessor for transforming the EHR text into clean tokenized text input essential for the success of the subsequently applied computational detection method. Our proposed NER system was ranked first in the MADE1.0 NLP Challenge for Detecting ADE information from EHR.

Third, we propose a multi-grained joint modelling approach for word-level and sentence-level information detection using deep learning. Existing ADE detection from text can be either fine-grained (ADE entity recognition) or coarse-grained (ADE assertive sentence classification), with limited efforts leveraging inter-dependencies among these two granularities. Moreover, in most attention-based neural network models for sentence classification only a single round of attention focusing on simple semantic information is applied for learning the importance of words and the overall representation of the sentence. We design a multi-grained joint deep network model MGADE to concurrently solve both ADE tasks MGADE takes advantage of their symbiotic relationship, with a transfer of knowledge between the two levels of granularity. Our dual-attention mechanism constructs multiple distinct representations of a sentence that capture both task-specific and semantic in-formation in the sentence, providing stronger emphasis on the key elements essential for sentence classification.

In several comprehensive experimental studies, namely, one for each part of this dissertation, we demonstrate the superiority of the proposed strategies over the state-of-the-art techniques with respect to precision, recall and F1-measure.

To my father-in-law Dr. K. S. Sharma

Acknowledgments

I sincerely thank my adviser, Prof. Elke A. Rundensteiner for guiding me in this journey with her immense knowledge and being supportive, playing a very important role in every step of my PhD studies, research and life. Her motivation and enthusiasm are contagious and she has been a constant source of inspiration for me to aim and achieve higher goals.

My special thank you goes to my dissertation committee members Dr. Xiangnan Kong, Dr. Eleanor T. Loiacono, and Dr. Marni Hall. I would like to express my gratitude to Dr. Xiangnan Kong for his excellent guidance, patience, insights and support for my research. I would like to thank Dr. Eleanor T. Loiacono, and Dr. Marni Hall for devoting their time and efforts to serve on my Ph.D. committee.

I sincerely thank my main collaborator, Dr. Xiao Qin, for numerous stimulating discussions. His critical comments significantly improved every publication we have accomplished together. It was a pleasure to work with him and I am looking forward to our future joint publications. I also thank Tabassum Kakar for continued research collaboration and being an excellent PhD companion. My thank you also goes to PhD colleagues Cansu Sen, Monica Lauren Tlachac (ML); REU students Vimig Socrates, Amber Wallace for productive collaboration. I would like to express my gratitude to all members of DSRG group for their support, useful discussions, and feedback.

I would like to thank Seeds of STEM: The Development of an Innovative Early Childhood STEM Curriculum, an IES funded project (Grant R305A150571) for supporting me as a research assistant during the first four years of my PhD studies. My special thank you goes to Dr. Mia Dubosarsky for being very supportive and encouraging me to pursue my PhD studies.

I am endlessly grateful to my parents and parents-in-laws for their encouragement and always being there for me. My husband, Dr. Venkat Kolluri and kids - Jaya & Shiva, they are my strength and I couldn't have embarked on this journey without their enduring support. This dissertation is truly devoted to them.

List of Publications during My Ph.D. Studies at WPI

Publications Contributing to this Dissertation

1. Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Xiangnan Kong, and Elke A. Rundensteiner. *A dual-attention network for joint named entity recognition and sentence classification of adverse drug events*. Empirical Methods in Natural Language Processing, Findings of EMNLP, 2020.

2. Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Cansu Sen, Elke A. Rundensteiner and Xiangnan Kong. *Adverse drug event detection from electronic health records using hierarchical recurrent neural networks with dual-level embedding*. Drug Safety. 42(1): pages 113–122, 2019.

3. Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Xiangnan Kong, Elke A. Rundensteiner: *Deep learning strategies for automatic detection of medication and adverse drug events from electronic health records*. American Medical Informatics Association Annual Symposium, Poster Paper. AMIA 2018.

4. Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Elke A. Rundensteiner, Xiangnan Kong: *Bidirectional lstm-crf for adverse drug event tagging in electronic health records*. In Proceedings of the 1st International Workshop on Medication and Adverse Drug Event Detection, vol 90 of Proceedings of Machine Learning Research, pages 48–56. 2018.

5. Susmitha Wunnava, Xiao Qin, Tabassum Kakar, M. L. Tlachac, Xiangnan Kong, Elke A. Rundensteiner, Sanjay K. Sahoo, Suranjan De. *Multi-layered Learning for information extraction from adverse drug event narratives*. In International Joint Conference on Biomedical Engineering Systems and Technologies. BIOSTEC 2018. Communications in Computer and Information Science, CCIS. vol 1024. pages 420–446. Springer, 2018.

6. Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Xiangnan Kong, Elke A. Rundensteiner, Sanjay K. Sahoo, and Suranjan De. *One size does not fit all: An ensemble approach towards information extraction from adverse drug event narratives*. In Proceedings of HEALTHINF, pages 176–188. Proceedings of HEALTHINF, 2018.

   Finalist to HEALTHINF 2018 best student paper award

7. Susmitha Wunnava. *Closing the gap: Structuring the unstructured adverse drug event narratives*. Doctoral Consortium on Biomedical Engineering Systems and Technologies. DCBIOSTEC 2018.

8. Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Vimig Socrates, Amber Wallace and Elke A. Rundensteiner. *Towards transforming FDA adverse event narratives into actionable structured data for improved pharmacovigilance*. Proceedings of the Symposium on Applied Computing. pages 777-782. SAC 2017.

Other Publications (Data Science Collaborations)

9. Xiao Qin, Cao Xiao, Tengfei Ma, Tabassum Kakar, **Susmitha Wunnava**, Xiangnan Kong, Elke Rundensteiner, and Fei Wang. *Supervised topic compositional neural language model for clinical narrative understanding*. IEEE International Conference on Big Data. 2020.

10. Xiao Qin, Tabassum Kakar, **Susmitha Wunnava**, Brian MacCarthy, Andrew Schade, Huy Quoc Tran, Brian Zylich, Elke A. Rundensteiner, Lane Harrison, Sanjay Sahoo, Suranjan De. *MeDIAR: Multi-Drug Adverse Reactions Analytics*. Proceedings of the International Conference on Data Engineering. Demo Paper. ICDE 2018

11. Xiao Qin, Tabassum Kakar, **Susmitha Wunnava**, Elke Rundensteiner, Cao Lei. *MARAS: Signaling Multi-Drug Adverse Reactions*. Proceedings of the International Conference on Knowledge Discovery and Data Mining. KDD 2017

Other Publications (Seeds of STEM, STEM Education Collaborations)

12. Sibuma, Bernadette, **Susmitha Wunnava**, Melissa-Sue John, Florencia Anggoro, and Mia Dubosarsky. *The impact of an integrated Pre-K STEM Curriculum on teachers' engineering content knowledge, self-efficacy, and teaching practices*. In 2018 IEEE Integrated STEM Education Conference (ISEC), pp. 224-227. IEEE, 2018.

13. Dubosarsky, Mia, Melissa Sue John, Florencia Anggoro, **Susmitha Wunnava**, and Ugur Celik. *Seeds of STEM: The development of a problem-based STEM curriculum for early childhood classrooms*. In Early Engineering Learning, pp. 249-269. Springer, Singapore, 2018.

14. John, Melissa-Sue, Bernadette Sibuma, **Susmitha Wunnava**, Florencia Anggoro, and Mia Dubosarsky. *An iterative participatory approach to developing an early childhood problem-based STEM curriculum*. European Journal of STEM Education, 3(3), 07. 2018.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background

1.1.1 Adverse Drug Events

Adverse Event (AE) according to World Health Organization (WHO) is any untoward medical occurrence that may appear during treatment with a pharmaceutical product but which does not necessarily have a causal relationship with the treatment [1]. An adverse drug event (ADE) is an injury resulting from medical intervention related to a drug[1] [2, 3, 4]. This includes medication errors, adverse drug reactions, allergic reactions, and overdoses. Adverse Drug Reactions (ADR), a subset of ADEs, that refers to an unexpected harm caused by the normal use of medication at the normal dosage [1]. ADRs are known to be a leading cause of death in the United States [5]. They are are also responsible for millions of hospitalizations and prolonged hospital stays costing billions of dollars to the health care systems[2] [6].

Clinical trials have limitations - they are restricted to certain populations, to a limited number of people, who may not always be representative of the population of all potential users of the drug, and conducted for a short period of time where it might not be possible to detect all potential ADEs. Many such ADEs are detected only during the post-marketing phase of the drug when it is used by a more diverse and much larger population than during clinical trials. Reducing the incidence of ADEs is extremely important for public health safety and reduced health-care costs.

1.1.2 Adverse Drug Event (ADE) Detection

Pharmacovigilance according to World Health Organization (WHO) is defined as the science and activities related to the detection, assessment, understanding and prevention of ADEs or other drug-related problems [1]. The specific aims of pharmacovigilance are to

[1] https:// health.gov/our-work/health-care-quality/adverse-drug-events

[2] https://www.fda.gov/drugs/drug-interactions-labeling/preventable-adverse-drug-reactions-focus-drug-interactions

improve patient care, public health and safety in relation to the use of medicines; contribute to the assessment of benefit, harm, effectiveness and risk of medicines, encourage their safe, rational and more effective use; and promote understanding, education and clinical training in pharmacovigilance and its effective communication to the public. Post-marketing safety data collection and risk assessment based on observational data are critical for evaluating and characterizing a product's risk profile and for making informed decisions on risk minimization.

Pharmacovigilance activities in the post-approval phase, principally involve the identification and evaluation of safety signals [7]. To oversee the safety and effectiveness of drugs in the post-marketing phase, different pharmacovigilance approaches are employed. Spontaneous reporting systems (SRS) such as the FDA Adverse Event Reporting System (FAERS) monitor ADE incidence reports submitted by consumers, healthcare professionals and drug manufacturers. These reports are reviewed by FDA staff to identify potential drug safety concerns and, when necessary, to recommend appropriate actions to improve product safety.

Real-time real-world data gathered during routine clinical visits such as Electronic Health Records (EHR), hospital discharge summaries and medical prescription claims have the ability to quantify the incidence or risk of ADEs, can identify patients at risk, and have the potential to provide more accurate and earlier ADE detection. For example, under the Sentinel Initiative [8], FDA is developing a new postmarketing surveillance system to monitor the safety of approved medical products in real time . The system currently uses administrative claims data, but FDA aspires to utilize EHR data for the same purpose in the later phases of the system development. Previous studies [9, 10] have proposed signal-detection strategies that combine data from FAERS and EHRs by requiring signaling in both sources, leading to improved accuracy of signal detection.

1.1.3 The Gap: Structuring Unstructured Narratives

In 2015, over 1.7 million of incidents had been reported to FAERS and the number has steadily been growing - making the drug review process increasingly challenging [11]. Similarly, following the Health Information Technology for Economic and Clinical Health Act (HITECH Act) legislation in 2009, there has been a significant increase in the number of health care institutions who have adopted EHRs. To effectively identify drug safety signals in a timely manner from the exploding amount of reports with limited human resources, the review processes are being enhanced by advanced data mining and visualization technologies [12, 13, 14]. However, most of these technologies rely on information organized in a structured format. Thus they require that the unstructured narratives first be processed and converted into structured information.

Although data sources such as SRS and EHR have both structured fields and unstructured free-form text fields, the unstructured narratives often contain information that is left blank in the structured fields. More importantly, these narratives are rich in detailed additional information regarding the ADE that could not be captured in rigorous structured menus. However, it is cumber-some, time consuming and error-prone to manually extract and enter the structured information from the narratives into the database. Therefore, efficient computational methods are required to automate the process of extracting information from the unstructured ADE narratives into structured format for advanced analytics. This is vital for timely detection, assessment and prevention of future incidents of ADEs while also making the clinical review process more productive and effective.

1.1.4 Computational Approaches for ADE Detection from Narratives

Information Extraction (IE) refers to the task of automatically extracting a predefined set of information categories also known as *named entities* or *concepts* from natural language text and classifying the relationships between the extracted entities. Named Entity Recognition (NER), fundamental step in the IE process, refers to the sub-task of identifying and

classifying textual mentions of *named entities* from the natural language text. Relation Extraction (RE) refers to the sub-task of classifying the associations or relationships between the *named entities* that are identified in the NER task. An IE application generally involves one or more of the sub-tasks.

Rule-based and machine learning are the two standard approaches to IE. Rule-based extraction methods are user-defined rules for pattern matching to the raw text to extract information [15]. However, manually creating rules covering all possible occurrences of information categories that need to be extracted requires human expertise and is labor intensive. Many of the IE sub-tasks such as NER can be formulated as classification problems. Machine learning-based extraction methods automatically learn from example patterns found in the training dataset and aim to classify information in the narratives. Machine learning algorithms can be supervised or unsupervised or semi-supervised. For supervised machine learning algorithms, the training examples have associated ground truth labels, i.e., the correct input-output pairs for each example from which the algorithm learns and then can predict new previously unseen examples. For unsupervised learning algorithms on the contrary, the associated ground truth labels are not available and hence the goal is to discover interesting patterns in the data using clustering algorithms.

Machine learning approaches for Natural Language Processing (NLP) to automatically extract ADE entities and their relations from the unstructured medical narratives has been an active area of research. For syntactic processing of the narratives, NLP tasks such as sentence segmentation, word tokenization, parts-of-speech tagging, text lemmatization, stop words removal, etc., are used in pre-processing of the text . For semantic processing of the narratives, traditional machine learning based text classification and sequence labeling techniques which assign categories, i.e., *named entities*, to text are used for entity extraction (NER) and relation extraction. These techniques however require heavy feature engineering resulting in a high-dimensional feature vectors that lead to over-fitting issues.

More recently, deep learning techniques with pre-trained word embeddings which are dense lower-dimensional feature vectors have been shown to be promising techniques for

sequence labeling and NER tasks due to their ability to learn from the context surrounding the words in a sequence. Attention mechanisms imitate human sight mechanism giving more weight to certain words or phrases over others in the text. Research has also shown that attention mechanisms have a greater impact on the neural networks for language translation [16, 17, 18] and especially with regard to ADR detection [19, 20].

1.2 State-of-the-Art

Significant amount of research has been conducted in ADE text classification using supervised machine learning approaches for NLP from unstructured medical narratives. We first describe the relationship of this dissertation to the broader class of ADE text classification approaches. Chapters 2, 3, and 4 discuss related work for each part of this dissertation in more detail. Broadly, we can categorize the approaches into the following areas of work.

1. **Document-level** classification refers to the task where a text document is assigned a class and classified as belonging to that specific class, for example the *ADE* class, based on word features extracted from the full text document.

   Machine Learning-based. Rochefort et al. [22] developed Support Vector Machine (SVM)-based document classifiers with bag of words feature vector to classify EHR documents which contain deep vein thrombosis and pulmonary embolism. Wang et al. [23] developed a logistic regression based document classifier with MeSH annotations and textual information from the abstracts and titles as feature vectors to identify PubMed articles which denoted a drug-ADE relationship. They demonstrated that the use of both semantic and syntactic features leads to improved performance for identifying ADE relationships.

   Deep Learning-based. Dev et al. [24] developed binary document classifier using logistic regression, random forests and 1-layer Long short-term memory (LSTM) to classify an adverse event case as serious vs. non-serious. For the feature vectors they have used Term-Frequency- Inverse-Document-Frequency (TF-IDF) and Binary

Count Vectorization (BCV) approaches from the information retrieval community to encode the words. They demonstrated that BCV showed significantly better performance in classifying the serious cases compared to the non-serious cases. While the deep learning was shown to outperform the baseline machine learning models, due to its lack of explainability the logistic and random forest were chosen over the 1-layer LSTM with word embeddings.

2. **Word-level** classification refers to the task where each word in the document is identified and classified as belonging to a certain pre-defined class, for example the *ADE* class. A majority of the existing NER systems for concept extraction are based on word-level detection.

   Machine Learning-based. Gurulingappa et al. [31] developed a machine-learning based relation extraction system based on Java Simple Relation Extraction (JSRE) a re-trainable and scalable classification platform that uses SVM with token-level textual features such as part-of-speech (POS) tags, lemmas for the identification and extraction of potential adverse drug event relations from MEDLINE case reports. Sampathkumar et al. [32] utilized a lexicon-based approach for detecting mentions of ADRs from online medical forums and then applied a Hidden Markov Model to detect relationships between drug-ADR pairs. Ramesh et al. [33] developed a biomedical named entity tagger using SVM with various categories of features such as syntactic, semantic, morphological, affix, negation, hedging and discourse connective features to improve the tagger performance in extracting medication and ADE information from FAERS narratives. Chapman et al. [34] developed a Conditional Random Field (CRF) model for NER and a random forest model for relation extraction with innovative feature engineering leveraging word embeddings with induced word clusters for dimensionality reduction. Wunnava et al. [35]

   Deep Learning-based. Jagannatha and Yu [36] have employed a bidirectional LSTM-CRF model with explicit modeling of pairwise potentials to label named entities

from electronic health records of cancer patients. Pandey et al. [19] proposed a bidirectional recurrent neural network (RNN) with attention mechanism to extract ADRs and classify the relationship between entities from Medline abstracts and EHR datasets by incorporating word embeddings using Word2Vec and GloVe, biomedical medical resources, drug- disease pairs from EHR text, abbreviations and positive-negative phrases compiled from the corpus. Li et al. [37] developed a joint model based on Bi-LSTM for simultaneously extracting drugs, diseases and adverse drug events mentions and their ADE relations using the shortest dependency path between the target entities in the dependency trees to help with the relation classification task. Li et al. [38] developed a combination of bidirectional LSTM (Bi-LSTM) and CRF network to recognize entities and extract relationship between entities from EHR using a multi-task learning (MTL) approach to improve the performance as compared to learning the tasks individually.

In our previous work, Wunnava et al. [39], we presented a three-layer deep learning architecture for identifying and labeling named entities from EHR, consisting of a Bi-LSTM layer for character-level encoding, a Bi-LSTM layer for word-level encoding, and a CRF layer for structured prediction. To better handle the noisy format of clinical notes, we built a rule-based sentence and word tokenizer leading to a better performance compared with using an off-the-shelf Natural Language Toolkit [40]. Our proposed system was ranked first for the NER task in the MADE1.0 NLP challenge for detecting medication and ADE related information from EHR. Dandala et al. [41] and Yang et al. [42] applied a similar BiLSTM-CRF structure for NER task and were placed second and third respectively in the challenge.

3. **Sentence-level** classification refers to the task where each sentence in the document is assigned a class based on the word features in the sentence to detect ADE and nonADE sentences.

   Machine Learning-based. Gurulingappa et al. [43] developed a maximum entropy

based classifier with morphological and syntactic textual features to identify and extract sentences that provide information about drug-related adverse effects from medical case reports. They also apply a lexicon-based approach to identify drug-names and conditions from the extracted ADE assertive sentences. Nikfarjam et al. [44] developed ADRMine, a CRF based system to extract ADRs mentions from user posts in social media using embedded cluster features by clustering words based on unsupervised pretrained word embeddings generated from unlabeled user posts in social media. The similarity modeling between the words showed significant improvement in the recall. Sarker and Gonzalez [45] developed Naïve Bayes (NB) and SVM based NLP systems for the automatic classification of ADR assertive text segments from Twitter and Daily Strength using semantic features such as topics, concepts, sentiments, and polarities. They have demonstrated that the integration of information from compatible corpora in the form of multi-corpus training can significantly improve classification performance especially when the data sets are heavily skewed.

Deep Learning-based. Huynh et al. [46] used Convolutional Neural Networks (CNN) augmented with Attention mechanism using pre-trained word embeddings to detect sentences describing ADEs. Their experiments however showed that CNN performed better as compared to CNNA (with attention). Tafti et al. [47] utilized a bigNN system based on feed forward artificial neural network to discover ADE sentences on massive datasets downloaded from PubMed Central and social media.

4. **Multi-level** classification which is a hybrid combination of two or more of the classification levels (i.e., document-level, word-level, sentence-level) in order to jointly classify information at multiple levels for multiple tasks in a single joint, end-to-end model. Multi-task learning (MTL) refers to training the model for multiple related tasks in parallel where information learned from a task can then be used to train another task [48]. Learning these tasks simultaneously may significantly improve the

performance as compared to learning the tasks individually. Zhang et al. [49] developed a multi-task learning model where they combined the word-level NER model with the document-level binary classification in a {Document-level + Word-level} model to both extract the Adverse Events (AE) from a case narrative and concurrently also classify the narrative document as serious or non-serious.

In our recent work, Wunnava et al. [50], we presented a dual-attention network for multi-grained ADE detection to jointly identify ADE entities and ADE assertive sentences from EHR narratives. Our model effectively supports knowledge sharing between the two levels of granularity, i.e., words and sentences, improving the overall quality of prediction on both tasks. Our solution features significant performance improvements over state-of-the-art models on both tasks.

Figure 1 shows the state-of-the-art landscape of ADE information extraction with respect to the classification goals and computational approaches used. Our research (blue colored boxes in Figure 1) is specifically focused on exploring Word-level, Sentence-level and {Word-level + Sentence-level} multi-level ADE information extraction using supervised machine learning and deep learning approaches. From an application viewpoint, the motivation behind the the {Word-level + Sentence-level} is that it closely mimics the workflow for processing a ADE narrative. One important property of human perception is that one does not tend to process a whole scene in its entirety at once. Instead humans focus attention selectively on parts of the visual space to acquire information when and where it is needed, and combine information from different fixations over time to build up an internal representation of the scene, guiding future eye movements and decision making [51]. As humans, we typically scan the text first, searching for keywords which relate to our search. Once we identify the keywords in a line or sentence, we read the entire sentence to understand the meaning of the sentence, that is, its entities and their relations.

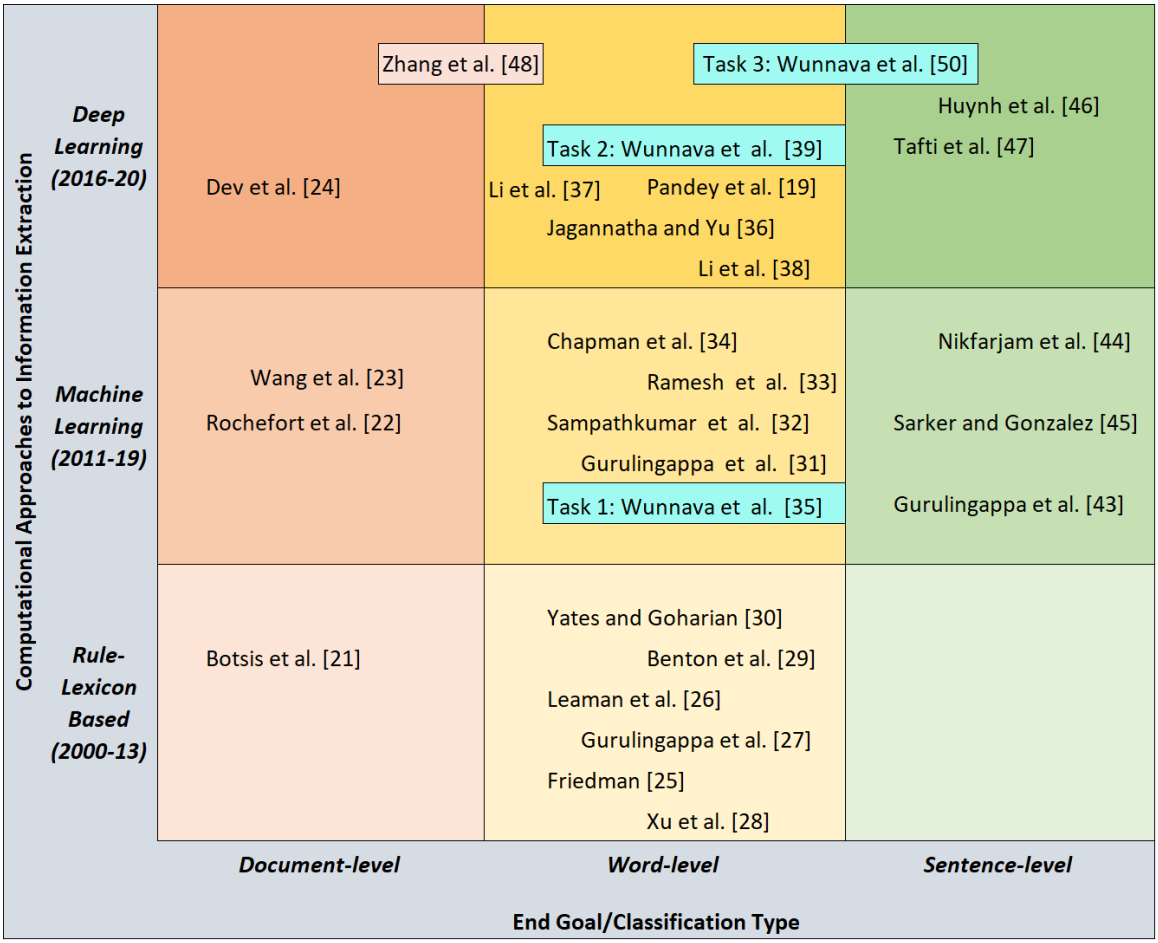


Figure 1: State-of-the-Art: Landscape of ADE Information Extraction.

1.3 Research Challenges

Although medical narratives contain valuable ADE information, processing them tends to be computationally difficult. A major hurdle with processing medical narratives is that the text is unstructured, comprised of different formats and styles depending upon the report source and does not always conform to grammar. Numerous challenges arise when detecting pre-defined information categories (*named entities*) such as *drugnames, dosage, duration, indications, ADE etc.*, from such narratives. In this dissertation, we focus on the following open problems.

1. Narratives contain medical and non-medical abbreviations, acronyms, numbers and misspelled words which make it difficult to recognize the critical information.

2. Entities can span across multiple words, about one to seven words per entity. Second, they could be expressed as a combination of entity-specific medical terms as well as non-medical descriptive text [35]. For instance, in the phrase "coronary artery disease related event prophylaxis", the words "related" and "event" are descriptive text while the rest are medical terms.

3. Narratives often contain information such as relevant medical history of the patient, family history and other diagnosis that is not related to ADE information categories. Semantic type ambiguity refers to the fact a word can have more than one interpretation depending on the context in which it is used. Semantic type ambiguity, also known as entity ambiguity is prevalent among certain named entities such as ADEs, Indications, other Signs & symptoms. Depending upon the context in which it is used, a word such as "headache" can be classified either an ADE, indication, a sign or a symptom. The context is also complicated by negated words.

4. Narratives are predominately composed of large chunks of texts with sparse relevant phrases specific to the named entities. For instance, in the MADE1.0 challenge training dataset, a de-identified dataset of EHR notes of 21 cancer patients from the University of Massachusetts Medical School [52], only 2.2% of the total annotations are for *ADE* and 4.7% for *Indication* while 50.2% are of *Sign & Symptom*. The data is highly skewed especially for the *ADE* entity leading to a class imbalanced dataset. It is well known that in such situations, the *ADE* entity being the minority class is not well learned and hence making it challenging to detect [53]. Supervised machine learning approaches need large amount of data specific to the named entities to be able to effectively learn the patterns in text and train the classification algorithms.

5. Existing ADE detection from text can be either fine-grained (i.e., identification of ADE named entities) or coarse-grained (i.e., identification of sentences describing ADEs), with limited efforts leveraging inter-dependencies among these two granularities. Unfortunately, when the interaction between these two extraction tasks is

ignored, we miss the opportunity of the transfer of knowledge between the entity and sentence prediction tasks.

1.4 Proposed Solutions

This dissertation focuses on tackling two distinct problem in the area of ADE detection. (i) Word-level information detection to identify the relevant ADE entities in the narrative and, (ii) Sentence-level information detection to identify ADE assertive sentence in the narrative. The research work can be broadly summarized into the following three tasks:

Task 1: Ensemble learning approach for word-level information detection from medical narratives. (Published works: [35, 54, 55])

The general problems of class imbalance and ensemble learning systems for classification have been studied in the literature [53]. However, in the context of NER from medical narratives, a collective approach to deal with both the class imbalance problem and the limitations of any one individual classification method has not been studied extensively. In this task, we developed a novel methodology called Tiered Ensemble Learning System with Diversity (TELS-D) to address the above challenges in NER. TELS-D involves four core steps:

1. Identify and generate diverse types of features from the textual narratives, including syntactic, semantic, morphological and contextual features as components of the extractors.

2. To address the class imbalance inherent in medical data used for machine learning training, we create a balanced training environment by applying undersampling techniques.

3. We generate an ensemble of diverse classifiers by training a set of heterogeneous learning algorithms in this balanced training environment.

4. We combine the intermediate results generated by each of the classifiers in the en-

semble to create a meta-training feature set. That is, the class predictions from the base learners are passed as input features to the meta-algorithm to learn what the correct output is, given the prediction patterns of the base learner.

5. We train a "*learner-over-learners*" meta-algorithm over the meta-level features to correctly learn and classify the named entities in the narratives.

6. Evaluate the TELS-D system with measures commonly used for evaluating classification methods such as Precision, Recall and F1-Measure at token-granularity level.

Task 2: Sequence labeling for word-level information detection from medical narratives using Deep learning. (Published works: [39, 56, 57])

In recent years, deep learning models such as Recurrent Neural Network (RNN) models [58] and especially Long Short-Term Memory (LSTM) [59] due to their ability to learn long-term dependencies from the context surrounding the words in a sequence have been shown to be promising techniques for sequence tagging and NER tasks. In this task, we developed a detection methodology by adapting a three layered deep learning RNN architecture of:

1. Bi-LSTM for character-level word representation to encode the morphological features of the medical terminology.

2. Bi-LSTM for capturing the contextual information of each word within a sentence.

3. Conditional Random Fields (CRF) for the final label prediction by also considering the surrounding words.

4. Experiment with different word embedding methods commonly used in word-level classification tasks to demonstrate the impact of an integrated usage of both domain-specific and general purpose pre-trained word embedding for detecting adverse drug events.

5. Develop a rule-based EHR text preprocessor for providing clean tokenized text input essential for the success of the proposed model.

6. Evaluate the system based on the strict matching in F1-score using exact phrase-level evaluation. The metrics for evaluating the systems are Precision, Recall, F1-score and the micro-averaged score which sums up the individual true positives, false positives, and false negatives of the system for different sets and the apply them to get the statistics.

Task 3: Multi-grained joint modelling approach for word-level and sentence-level information detection using deep learning (Published works: [50])

Most previous approaches tackle multi-grained (word-level and sentence-level) information detection problem in a pipe-lined manner, where the inter-dependencies among the two levels of granularities are not well captured. Moreover, previous approaches typically apply only a single round of attention focusing on simple semantic information and consequently risks losing important cues. In this task, to tackle the above challenges, we design a dual-attention based neural network model for multi-grained joint learning, called MGADE, that jointly identifies both ADE entities and ADE assertive sentences. Our key contributions of this work are:

1. We designed an end-to-end joint model that leverages the mutually beneficial relationships between the word-level and sentence-level information detection tasks.

2. We introduce a novel method for generating and pooling multiple attention mechanisms to produce informative sentence-level representations.

3. We compute dual-attention weighted sentence-level representations to capture both task-specific and semantic information in a sentence, providing stronger emphasis on key elements essential for sentence classification.

4. We further utilize our dual-attention mechanism that can assign importance weight to the words in the input text to provide an intuitive explanation of MGADE model

and insights on its strengths and weaknesses.

5. We introduce an augmented sentence-level representation comprised of predicted entity labels for better capturing the word-level label distribution and word dependencies within the sentence.

6. Evaluate the MGADE system using Precision, Recall and F1-score. We compare MGADE model with state-of-art methods for the ADE entity recognition and ADE sentence classification tasks.

1.5 Use Cases for ADE Information Extraction

In the context of the post-marketing drug safety surveillance process, we describe two important use cases for ADE information extraction from the EHRs:

1. **Signal Characterization from EHR.** According to the CIOMS VIII Working Group [60], a signal refers to "information that arises from one or multiple sources , which suggests a new potentially causal association, or a new aspect of a known association, between an intervention and an event or set of related events, either adverse or beneficial, that is judged to be of sufficient likelihood to justify verificatory action" . Safety signal detection is both an iterative and dynamic process. It is in the best interest of public health to integrate and understand evidence from all possibly relevant information sources on drug safety [61]. Once the signal is identified, a standardized case definition is formulated with a set of uniformly applied criteria for determining whether a person should be identified as having a particular disease, injury, or other health condition. In the event that a safety signal warrants further investigation, data sources such as spontaneous reporting systems, structured longitudinal observational healthcare databases (such as, EHR, administrative claim systems), unstructured/free-text sources (such as, EHR/clinical narratives, scientific literature and patient-generated content) are used for additional characterization (prevalence & relevance) of the signal [7, 62].

The unstructured EHR narratives produced by healthcare professionals contain rich documentation of clinical conditions, treatments, and patient history. However, they are not easily accessible by computerized systems, and require manual review and manual identification of the important information which is time-consuming and cumbersome. The full potential of EHR data cannot be captured unless there is a reliable stream-lined way to extract relevant information. Our machine learning and NLP based information extraction methods could be utilized for additional characterization of the signal from EHR narratives. This aids in better understanding and analysis of the prevalence & relevance of signals (ADE) such as, identifying populations subsets, comorbid conditions, demographic characteristics of patients associated with the events, drug exposure duration, dosage, mode of drug administration, severity of reaction and outcome, concomitant medications.

2. **Evaluate vaccine safety in real-world settings.** With regard to drug safety surveillance, EHR databases have been commonly used to confirm or refute potential signals detected initially by SRS, including vaccine-related signals [61]. Rare but serious adverse events associated with vaccines are often nearly impossible to detect in pre-approval studies and require monitoring after introduction of the agent in large populations [63]. The longitudinal nature of routinely-collected EHR data may allow identification of adverse events that have a long delay between exposure and clinical manifestations [61].

   Especially, with the emergence and ongoing crisis of the Coronavirus disease 2019 (COVID-19 global pandemic), several efforts are underway to create effective vaccines for immunization and controlling the spread of the virus. In the Operation Warp Speed's Strategy and Approach (OWS) [64] document entitled "From the Factory to the Frontlines," [1] the Department of Health and Human Services (HHS) and the Department of Defense (DOD) stated that, because some technologies have lim-

[1] https://www.hhs.gov/sites/default/files/strategy-for-distributing-covid-19-vaccine.pdf

ited previous data on safety in humans, the long-term safety of these vaccines will be carefully assessed using pharmacovigilance surveillance and Phase 4 (post-licensure) clinical trials. The RCTs in the pre-approval phases cannot detect all possible/ potential vaccine related adverse events. Once these vaccines are approved and administered to larger populations, there is a need for a rapid, reliable way to better understand the safety of the vaccines. Our machine learning and NLP based information extraction methods method could be utilized for vaccine safety surveillance using EHR narratives. EHR data may be able to highlight new risks associated with the vaccine, as well as adverse events that have high incidence rates and events that are not pharmacologically predictable [61].

1.6 Dissertation Outline

This dissertation is organized as follows.

- Chapter 2 presents our work on our ensemble learning approach for word-level ADE information detection.
- Chapter 3 presents our work on our deep learning based sequence labeling architecture for word-level ADE information detection.
- Chapter 4 presents our work on our joint modelling approach for multi-grained (word-level and sentence-level) ADE information detection using deep learning.
- Chapter 5 concludes this dissertation and proposes future research directions.

2 Ensemble learning approach for word-level information detection

In this chapter, we propose a balanced, under-sampled bagging strategy that is dependent on the imbalance level to overcome the class imbalance problem. Second, we present an ensemble of heterogeneous recognizers approach that leverages a novel ensemble combiner. Our experimental results show that for biomedical text datasets: (i) a balanced learning environment along with an Ensemble of Heterogeneous Classifiers constantly improves the performance over individual base learners and, (ii) stacking-based ensemble combiner methods outperform simple Majority Voting by 0.30 F-measure. This work was published at HEALTHINF [35].

2.1 Introduction

2.1.1 Motivation and Background

Adverse Drug Reactions (ADRs) correspond to an unwanted and often extremely dangerous effect caused by the administration of drugs. ADRs unrevealed during the clinical trials are one of the leading causes of death worldwide [5]. To oversee the safety and effectiveness of the drugs in the post marketing phase, surveillance systems such as FDA Adverse Event Reporting System (FAERS) monitor the ADR incidences submitted by *consumers*, *healthcare professionals* and *drug manufacturers*. These reports are reviewed by FDA staff to identify potential drug safety concerns and, when necessary, to recommend appropriate actions to improve product safety.

In 2015, over 1.7 million of incidents are reported to FAERS and the number is growing making the drug review process more challenging [11]. To effectively identify drug safety signals in a timely manner from the exploding amount of reports with limited human resources, the reviewing processes are enhanced by advanced data mining and visualization technologies [12, 13, 14]. However, most of these technologies rely on information

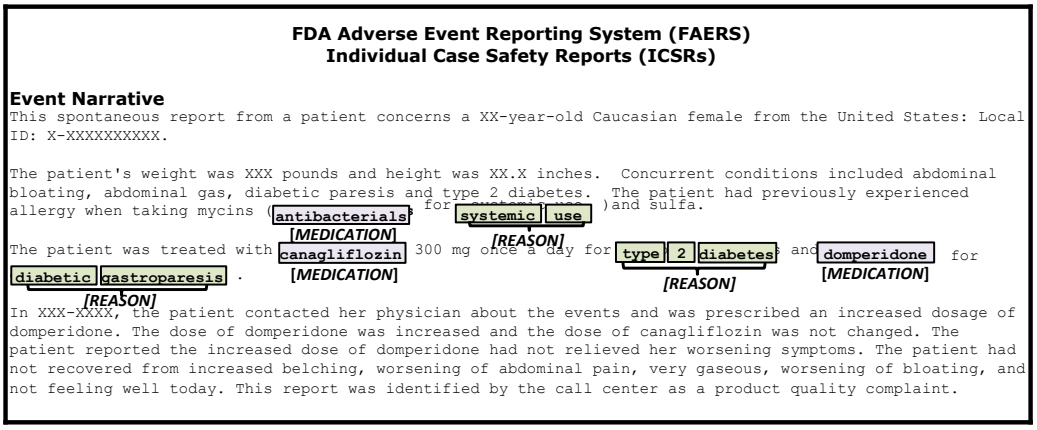


Figure 2: A sample FAERS report highlighting detailed information on the ADR incident within the narrative

organized in structured format where the unstructured text has to be first processed and converted into structured information.

Although the original report has structured fields, the unstructured narratives in the MedWatch form used for reporting an adverse event (Illustrated in Fig. 3) often contain information that is left blank in the structured fields. More importantly, these narratives are rich in detailed information regarding the adverse event as shown in Fig.2. Automatically extracting information from the unstructured ADR report narratives into structured format is critical for advanced analytics and vital for timely detection, assessment and prevention of future incidents of ADRs. In this study, we focus on the Named Entity Recognition (NER) – a fundamental task in this process, to classify the information categories in the narratives.

A major hurdle with biomedical narratives especially with processing medical reports is that the text is unstructured, comprised of different formats and styles depending upon the report source. First, a named entity phrase could be expressed as a combination of entity-specific medical terms as well as non-medical descriptive text. For instance, in the named entity phrase "*coronary artery disease related event prophylaxis*", the words "related" and "event" are descriptive text while the rest are medical terms. Named entity phrases such as these can cause ambiguity even during the manual annotation process. Second,

the narratives are predominately composed of large chunks of texts with sparse relevant phrases specific to the named entities.

Given above observations, it is a common protocol to engage multiple expert annotators specializing in different types of biomedical text and specific types of named entity to recognize and tag phrases and, then as a final step combine their expert opinions to come to an inter-expert agreement for determining the final output. As shown in our experiments, this problem persists when it comes to automatically recognizing entities through computational approaches. A named entity recognizer for biomedical text is usually designed for specific text type or entity type where a generic approach will almost certainly fail the domain specific task. Recently, many biomedical NER systems [28, 65, 66, 67] and frameworks [68] have been proposed customized for specific domain and entity type. To the best of our knowledge, there is no study today on how to automatically adapt and integrate the strength of a relevant and yet diverse set of named entity recognizers to tackle a new domain specific NER task.

2.2 Related Work

Existing approaches to biomedical NER can be categorized into rule-based, machine learning based and hybrid methods.

The rule-based methods leverage user-defined pattern matching rules supported with semantic knowledge resources. MedLEE [69] and MedEx [28] are rule-based systems that use a medical knowledge base and a linguistic approach to extract relevant medical information from clinical text. While rule-based systems perform well on identifying known patterns, they are limited in their ability to generalize. They thus fail to identify unknown words and patterns.

Machine learning based methods learn from features extracted from words and thus have a better generalization ability compared to rule-based methods. However, they require large annotated corpora for training. [70] demonstrated that machine learning approaches can outperform rule-based systems for assertion classification in clinical text. [33]

developed a biomedical named entity tagger using Support Vector Machines (SVM) to extract medication and ADR information from FAERS narratives. [71] used Conditional Random Fields (CRF) to label diseases and disorders in clinical sentences. [72] used a Maximum Entropy model to extract relevant medical information. [73] used Recurrent Neural Networks to extract medical events from Electronic Health Records (EHR) and showed that they significantly outperformed the CRF models.

Hybrid approaches that utilize both rule-based and machine learning methods have also began to be explored. [74] developed an SVM based method that utilizes the semantic tags of the words obtained from MedEx as features to recognize medication-related entities from discharge summaries.

2.2.1 Challenges of Entity Recognition using Machine Learning

The focus of our research is on supervised machine learning methods for biomedical NER and classification. In particular, we focus on a two-class, binary classification task to recognize and classify named entities. Despite its value and significance, biomedical NER and classification is a more challenging task due to the specific characteristics of the task. Two of the most critical challenges are:

1. *Lack of Positive Class Instances & Class Imbalance*: One problem in classifying named entities in biomedical text especially clinical text is that the data in the training dataset is predominately composed of non-medical text with only a small percentage of entity-specific medical text leading to highly skewed and imbalanced class distributions. Usually, the positive class, i.e., the class of interest that represents the named entity, has very few instances and is in a stark minority compared to the negative class (e.g., reason vs non-reason instances in the narratives, see Fig. 2).

   Research [75, 76] has found that, learning on imbalanced training datasets can cause a significant deterioration in the performance of the supervised machine learning methods, particularly when classifying instances belonging to the under-represented

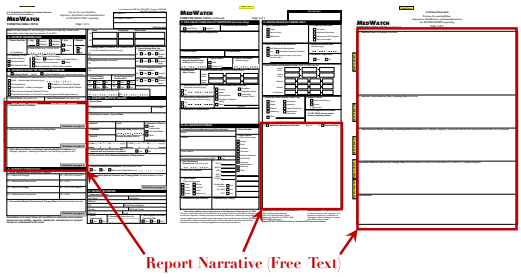


Figure 3: FAERS report – Medwatch 3500A

class.

2. *Lack of a Single Best Performing Classification Method*: It is challenging to choose the appropriate learning algorithm to train and classify the new instances. Conventional approaches to biomedical NER tend to use a single machine learning method such as Support Vector Machines (SVM), Conditional Random Fields (CRF), Maximum Entropy (ME) [77] classify named entities in the text. Each of these methods have some advantages over the others and differs significantly in their performances in classifying the named entities. [66] shows that the teams that used different supervised machine learning methods on the same dataset obtained significantly different results from one another. Additionally, the performances of a single system across the various named entities is shown to differ. [66] concluded that although the state-of-the-art NLP systems perform well in extracting some of the named entities (such as medication, dosages), while other entities (duration, reason for administration) have shown to be very challenging.

2.2.2 The Scope of this Work

The general problems of class imbalance and ensemble learning systems for classification have been studied in the literature [53]. However, in the context of biomedical NER, a collective approach to deal with both the class imbalance problem and the limitations of any one individual classification method has not been studied extensively. In this paper, we thus design a novel methodology called Tiered Ensemble Learning System with Diversity

(TELS-D) to address the above challenges in NER. TELS-D involves four core steps: 1) To address the class imbalance inherent in medical data used for machine learning training, we create a balanced training environment by applying undersampling techniques. 2) We generate an ensemble of diverse classifiers by training a set of heterogeneous learning algorithms in this balanced training environment. 3) We combine the intermediate results generated by each of the classifiers in the ensemble to create a meta-training feature set. 4) We train a "*learner-over-learners*" meta-algorithm over the meta-level features to correctly learn and classify the named entities in the narratives.

To evaluate our model, we perform comprehensive experiments on biomedical reports datasets. Our experiments demonstrate that our proposed methodology TELS-D outperforms the individual learners in the ensemble. TELS-D achieves a higher accuracy of 0.52 F-measure compared to any of the individual classifiers with F-measure ranging from 0.22-0.33, in recognizing the relevant information categories from the narratives.

2.3 Methodology

2.3.1 The Data Set

The FDA FAERS Adverse Event Report Narratives. The FDA Adverse Event Reporting System (FAERS) is a database that contains information on adverse events and medication errors in the form of reports submitted to the FDA from various sources such as patients, medical professionals and drug manufacturers. A report contains both a structured section of content followed by some free-form text. Fig. 3 depicts an example of MedWatch report form supported by FAERS. As many studies indicate [78], the narrative can be either supplementary material to the structured fields or in many cases reporters tend to provide a detailed narrative in the unstructured format without taking the effort to fill in all the structured fields. Therefore, there is a need for identifying information related to the adverse event case from the free text in order to collect all relevant knowledge about the case in structured and thus a easy processable format.

Table 1: Statistics for the datasets

	FAERS	i2b2
#Reports	16	242
#Sentences	678	8,050
#All Words	6,116	67,074
#Reason Words	NA	1,881

In this study, we aim to identify one important piece of knowledge, namely the *reason* thought to be the cause of the administration of the medication as per the FAERS report narrative. While we work with 925 FAERS reports, they are unlabeled and not redacted and therefore not available to the general public due to patient's privacy concerns. In addition, we also work with 16 redacted reports provided by the FDA as briefly described in Table 1.

Data set of Annotated Patient Discharge Summaries by Partners Healthcare. To assure reproducibility, we also work with the publicly available data set from the 2009 Medication Extraction Challenge from the Third i2b2 Workshop on Natural Language Processing Challenges for Clinical Records [79, 66]. The data set consists of annotated patient discharge summaries provided by Partners Healthcare. As part of the challenge, 696 reports were released for training out of which 17 reports were annotated by the i2b2 organizers. An additional 251 reports were released as the testing data set and were annotated by the participating teams. Annotated entities include *medication name*, *dosage*, *mode*, *frequency*, *duration*, and *reason for administration*. We work with 242 annotated reports (9 from the annotated training set and 233 from the testing set) as described in Table 1.

In this work, we focus on identifying the *reason* entity for the administration of drug from these discharge summaries. First, the *reason* entity has routinely been pointed out as one of the important fields yet among the hardest to recognize and extract due to its diversity and often not well scoped vocabulary [66, 72]. The original dataset features a heavy class imbalance with respect to the *reason* type. That is, tokens labeled as belonging to the *reason* class represent about 1% of all the tokens in these reports. Since the goal of this study is to develop an information extraction strategy that successfully identifies the *reason for administration* from the text, we focus on the narrative section of each report.

2.3.2 Data Pre-Processing

Data pre-processing is vital for converting the raw textual data into a processable format suitable for the natural language processing. We use following steps to pre-process each report in the corpus:

1. *Sentence Segmentation*: Each report is split into sentences to decompose the structure.
2. *Word Tokenization*: Each sentence is split into tokens (words) as this is our unit of processing.
3. *Punctuation Removal*: All tokens that represent punctuations are removed.

2.3.3 Feature Extraction

A rich set of features are needed for machine learning to learn the meaning of tokens. For each word token obtained from the preprocessing module we generate the following feature sets:

1. *Word Features*: The token is converted into a bag-of-words representation based on the vocabulary of the entire corpus. To generate the vocabulary, words in the corpus are converted to lowercase and stemmed using the NLTK Porter Stemmer [80].
2. *Syntactic Features*: A constituency parse tree is created using Charniak-Johnson parser [81]. Each token is tagged with its respective parts-of-speech (POS) and lexical categories.
3. *Semantic Features*: Semantic categories of the word are then obtained through lexicon lookup from medication lexicons, side effect lexicons (such as SIDER) [82] as well as UMLS Metamap [65].
4. *Context Features*: Words adjacent to the token in the narrative provide the context in which the token is actually used. This feature is helpful to differentiate when a token falls into one of two different sections of a report and thus labeled differently. A

context window size five words i.e,. two words before and two words after the token are coded using bag-of-words representation. A boolean value is a binary flag that indicates whether this token occurs before or after certain so called "trigger words". We identify trigger words that may indicate the presence of the named entity *reason*.

5. *Morphological*: The suffix and prefix of up to 3 characters within this word. For example: 1) words with prefix of "dys" indicate something is abnormal, such as dyspnea, 2) words with a suffix of "ing" may indicate a condition or symptom, such as bloating.

6. *Orthographic*: Boolean values are used to indicate if this word contains capital letters, digits, special characters, etc.

2.3.4 Base Machine Learning Models

After each token has been characterized by descriptive features by the above step, the tokens in the form of feature vector along with their associated label indicating its class type (*reason* or *non-reason*) are then used to train the models.

Different machine learning models have their own set of assumptions and way of modeling the data, resulting in its pros and cons in the classification task. In our study, we assume that different models are able to capture different aspects of the data and having them compliment each other in an assembly fashion will achieve better accuracy than any of them working individually. We build our base classifiers using multiple popular machine learning models, namely, Decision Tree (DT), Logistic Regression (LR) and Support Vector Machine (SVM) [83] (Illustrated in Fig. **??**).

2.3.5 Ensemble of Classifiers

Ensemble of classifiers is a group of diverse classifiers whose classification recommendations are aggregated to achieve more accurate classification [83, 84]. The goal of an ensemble system is to combine the results of many diverse classifiers into a single consensus re-

sult that outperforms any one of the individual classifiers by reducing their generalization error and thus their misclassification rate. The generalization error of the ensemble system tends to be lower than that of the individual classifiers when there is sufficient diversity in the ensemble where the base learners have different prediction accuracy on different instances. This makes the assumption that the base learners are better performing than random guessing. They have an accuracy greater than 50% [85].

2.3.6 Ensemble Generation: Model Diversity

1. Heterogeneous Learning Methods: One approach to generating a diverse set of classifiers is to train different learning methods on the same training set. If the performance of each of these methods varies significantly, then the results obtained are diverse in nature. Then to overcome the limitations of each learning algorithm while taking advantage of their respective strengths is to combine the classifiers into an ensemble of classifiers. In this study, we thus follow this methodology and create an ensemble of models obtained with the SVM, LR and DT learning methods. Our experiments (see Sec.2.4.8) confirm that an ensemble of these base classifiers outperforms any one of them.

2. Heterogeneous Training Datasets: Another common approach to generating a diverse set of classifiers is to create different subsets of the original training dataset and then to train a single learning method on each of the subsets from the training data set. Bagging [86] and Boosting [87] are examples of algorithms that tackle the generation of collection of classifiers by sub-setting the original dataset. However, given that our data set suffers from a heavy class imbalance problem and further the data size in terms of relevant tokens is limited, boosting or bagging, which further reduce the data to smaller subsets of data, are not suitable design options.

2.3.7 Ensemble Combination: Model Assembly

The combination method that combines the results of the diverse learning methods in the ensemble to obtain one aggregated consensus result can be achieved through different techniques. The most commonly used technique is Majority Voting (MV), that is, selecting as result the class that receives the highest votes from all the individual learning methods by simple counting. It can be simple or weighted voting where base learners are given different weights. In either case, the average is taken.

Another technique is Stacked Generalization [88] or in short Stacking, which is a *learning over learners* method to procure the final result. Stacking is a meta-learning algorithm where the class predictions from the base learners are passed as input data to the meta-algorithm to learn what the correct output is, given the prediction patterns of the base learner. In our study, we experiment with both Majority Voting and Stacking techniques as model combiners. Ultimately, we demonstrate that Stacking method outperforms Majority Voting and therefore is a promising strategy to adopt for combining the models into an ensemble.

2.3.8 Strategies for Addressing the Class Imbalance Problem

In biomedical named entity recognition tasks, often the training datasets used are very skewed, that is, they suffer from a heavy class imbalance [89]. Class imbalance occurs when one of the two classes, usually the class of interest, the positive class is in stark minority and the negative class is in majority. The performance of machine learning methods trained over such class-imbalanced datasets tend to be greatly affected by such class imbalance. In particular, this tends to result in the minority class not being well learned and hence misclassified most of the time. Class imbalance can influence the performance of the ML method by favoring the majority negative class. Approaches to deal with class-imbalanced datasets are described next.

2.3.9 Balancing with Class Weights

One common method is to balance the class weights within the classifier, thereby giving more importance (or weight) to the errors of the minority class. Higher class-weight puts more emphasis on the minority class. That is, it penalizes the model for making classification mistakes on instances of the minority class during training. These penalties bias the model to pay more attention to the minority class.

Usually, in the case of balanced datasets both classes are given an equal weight of one. In imbalanced datasets however, the class weights can be balanced by performing a grid search with different class weight combinations to find the optimal class weights. These weights are then passed to the learning method to bias the decision making process of the learning method.

2.3.10 Balancing with Class Instances

Another approach to minimize the effect of class imbalance is to re-sample the original training dataset to create a new modified training dataset that has a balanced class distribution. Random over-sampling and random under-sampling are both common re-sampling techniques [90]. In both cases, the objective is to decrease the effect of the highly skewed class distribution by creating a balance between the number of majority and minority class instances. This then enables the classifier to give equal importance to both classes during the training phase.

However, both techniques have limitations. While with under-sampling there is a possibility of throwing away important instances, with over-sampling we tend to increase the size of the training dataset. In this study, since our training dataset is already large and high dimensional, we choose to re-sample the dataset with the random under-sampling technique.

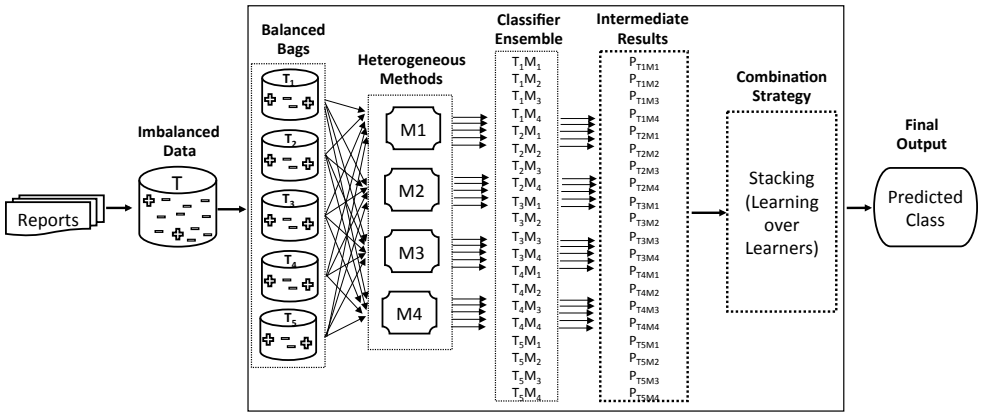


Figure 4: TELS-D tiered ensemble learning system with diversity

2.3.11 Balancing with Classifier Ensembles

Yet another approach to deal with class imbalance is to use ensemble methods to generate a classifier ensemble that can create a balanced learning environment for the learning algorithm [91]. Under-Bagging [92] and Over-Bagging [93] are examples of ensemble techniques, that deal with class imbalance in the learning phase through a combination of data re-sampling and bagging approaches, known as "balanced bagging".

To the best of our knowledge, with the above existing methods, the diversity in the ensemble is usually generated through training one homogeneous learning algorithm on all balanced subsets of the training data. The results from the classifier ensemble are aggregated using the Majority Voting combination method. In this study, although we will employ the basic idea of "balanced bagging", we will also extend it to train a diverse set of heterogeneous learning algorithms in parallel.

2.3.12 Tiered Ensemble Learning System with Diversity

In this study, to address the two challenges of (1) class imbalance and (2) the lack of a single best performing method, we propose a novel integrated approach to create a balanced learning environment. This strategy combines balanced resampling techniques with an ensemble of heterogeneous classifiers into one methodology. Our approach called **T**iered

Ensemble **L**earning **S**ystem with **D**iversity (**TELS-D**), effectively deals with the class imbalance problem in the data through a balanced under-sampled bagging approach, while also addressing the limitations of using a single learning method by training multiple heterogeneous learning methods on the under-sampled subsets in parallel.

The imbalance level in a dataset is defined as the ratio of the number of majority negative class instances to the number of minority positive class instances (Eq.1). It indicates how many times the majority class is greater than the minority class.

$$\text{Imbalance Level (IM)} = \frac{\text{\# Negative class tokens}}{\text{\# Positive class tokens}} \tag{1}$$

Based on the imbalance level of a dataset, we create multiple smaller subsets of the original dataset that each individually exhibit a balanced class distribution. That is, each smaller balanced subset takes *all* of the available positive class instances while working with only an equal number of negative class instances, i.e., a subset of the available negative class instances. The purpose here is to learn the features inherent in the positive class (the class of interest) without getting overwhelmed by the majority negative class instances and their typical characteristics. The number of subsets to form is determined by the imbalance level in the dataset. For example, in Fig. 4, the negative class is five times larger than the positive class. Hence, the original unbalanced training dataset (DB) is split into five smaller balanced subsets henceforth called "balanced bags" (BB) while ensuring that we do not discard any instances from either classes, i.e., $\cap_{i=1}^{5}$ BB $\neq \emptyset$ and $\cup_{i=1}^{5}$ BB $= DB$

For example, if the imbalance level in the dataset is N, then we create N (N $>$ 1) balanced training sets, BB. If we have M (M $>$ 1) base learning methods, we train T $=$ N $\times$ M base learners in the first layer of the ensemble. So, instead of creating an ensemble of just N diverse models (Sec. 2.3.6-2) or just M diverse models (Sec. 2.3.6-1), with our proposed TELS-D strategy we create a collection of T diverse models.

The advantage of TELS-D approach is that we generate more diversity in the ensemble while balancing the class distribution. With more diverse base learners, each one of the

T base classifiers will make different errors on different instances. We then combine the results from these T diverse base learners to form an input for the second layer stacking meta-algorithm. This gives the meta-learner an opportunity to learn the patterns to predict the correct class - thereby reducing the total error.

2.3.13 Evaluation Criteria

We adopt the criteria commonly used for evaluating classification methods, but now adapt them to apply to the token-granularity level. That is, we measure both the *Precision* and *Recall* as described below to determine whether or not the learning models sufficiently capture the classifications of the positive class.

$$\text{Precision (P)} = \frac{\text{\# Correctly predicted positive tokens}}{\text{\# Total predicted positive tokens}} \tag{2}$$

$$\text{Recall (R)} = \frac{\text{\# Correctly predicted positive tokens}}{\text{\# Total real positive tokens}} \tag{3}$$

Our goal is to achieve high precision (lesser false positives) and high recall (more true positives). Thus, *F-measure*, defined below, gives a balance between both precision and recall measures, thereby balancing the accuracy of both positive and negative predictions. Hence, F-measure is a commonly accepted measure to evaluate the performance of learning methods.

$$\text{F-measure (F1)} = \frac{2(P\ x\ R)}{(P\ +\ R)} \tag{4}$$

2.4 Experimental Results

2.4.1 Experimental Setup

Data Sets In this study, to build and evaluate our classification approaches we have used the data set of annotated patient discharge summaries from i2b2 (Sec.2.3.1) that has been

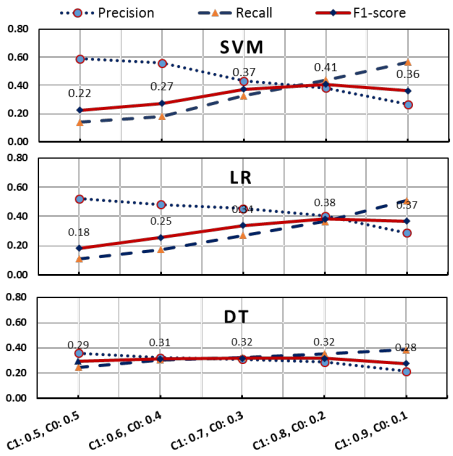


Figure 5: Grid search results for balancing class weight on target *Reason)* in *i2b2* dataset

augmented with ground truth labels which are needed for supervised machine learning strategies. Holdout test set approach is adopted with a 90/10 split. The i2b2 corpus, the 242 reports used in this study (Table 1) are split accordingly where 90% of the reports (217 reports) are randomly selected for training and building our proposed model and the remaining 10% (25 reports) are used as the holdout for subsequent testing to evaluate the effectiveness of our methods. In this section we discuss our empirical results on this holdout test set. Additionally, we have experimented with the 16 FAERS reports as a second test set (Table 1). Due to lack of ground truth labels for these FAERS reports, we manually evaluated the results and present a case study as part of our results discussion.

Parameter Tuning Base learners such as SVM and LR must be tuned first and parameters are used to do so. Therefore, we have used SVM with a linear kernel function and LR with a *c-value* of 1.0. These values were the best parameters we obtained after testing with *c-values* (0.001, 0.01, 0.1, 1, 10) using 10-fold cross-validation [94]. The *c-value* controls the trade off between model complexity and misclassified instances. We have used decision tree with *best* split at each node strategy and *gini* to measure the quality of the split [85]. For selecting the optimal class_weight setting, we performed a systematic grid search with a set of class weights for each class using 10-fold cross-validation. The effect of balancing different *class_weight* values on individual learning methods (SVM/LR/DT) is depicted in

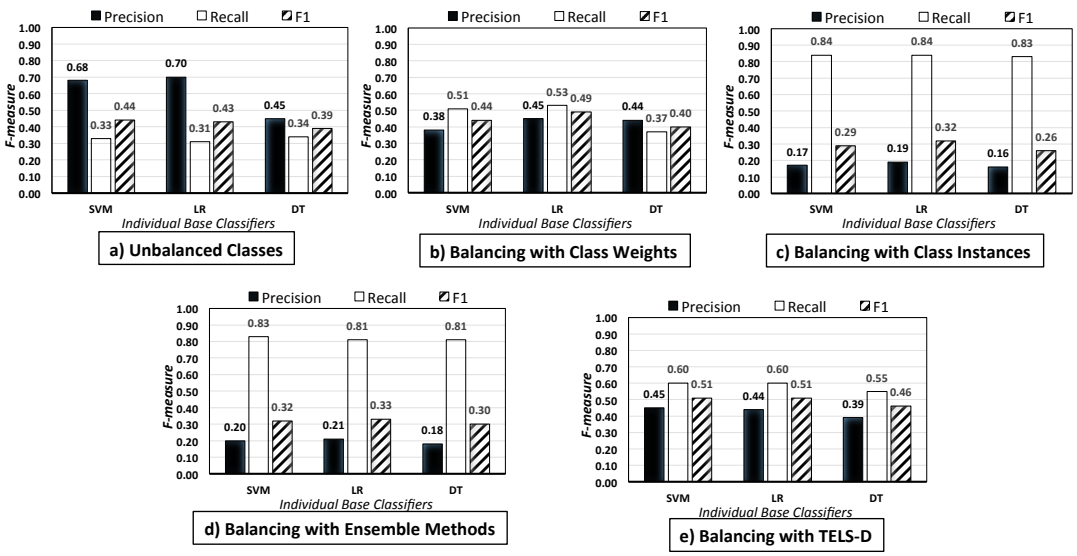


Figure 6: The precision, recall and F1-score of different classification strategies

Fig.5. This experiment shows that for the three base learners, the precision and recall are balanced with a higher F-Measure at a *class weight* $\{C1 : 0.8, C0 : 0.2\}$ setting, where C1 denotes the class *reason* and C0 denotes the class *non-reason*. We thus set the *class weight* to $\{C1 : 0.8, C0 : 0.2\}$ throughout the rest of our experiments where we balance the class weights within the learning methods.

2.4.2 Classification with Unbalanced Class Distribution

This experiment is conducted to obtain a baseline to compare the different approaches explained in Sec.2.3.8. The individual base learners are trained on the original training set (DB) without balancing the class weights or instances (Fig.6 (a)) to see the effect of skewed class distribution.

In this experiment, the precision P is much higher than the recall R for all base learners especially for SVM (P:0.68/ R:0.33) and LR (P:0.70/ R:0.31). High precision and low recall implies very few tokens were predicted as belonging to *reason* class, but most of them are correct predictions when compared against ground truth labels. This is expected due to the class imbalance, with the majority of the tokens being *non-reason* labels in the training phase. Thus the base classifiers are biased towards the *non-reason* class and tend to mis-

classify most tokens in the minority *reason* class.

2.4.3 Balancing with Class Weights

This experiment is conducted to evaluate the effectiveness of the strategy of balancing class weights to address the data imbalance problem. The *class weight* parameter is set to $\{C1 : 0.8, C0 : 0.2\}$ in the individual base learners. The base learners are then trained on the original training set (DB) (Fig.6 (b)).

In this experiment, the recall is now higher than the precision for two base learners, SVM (P:0.38/ R:0.51) and LR (P:0.45/ R:0.53). High recall and low precision implies many tokens were predicted as belonging to *reason* class. However, most of them are incorrect predictions when compared against ground truth labels. This is expected because, in order to deal with class imbalance during the training phase, we had set the class weights within the base learners such that the minority *reason* class is given more weight. Hence this tips the classifier learning bias towards the minority *reason* class. In contrast to the Unbalanced experimental results (Sec.2.4.2), this now had led to more of the majority *non-reason* class tokens being misclassified as *reason* class. The evaluation metrics of DT (P:0.44/ R:0.37/ F1:0.40) are similar to the unbalanced experimental results (Sec.2.4.2).

2.4.4 Balancing with Class Instances

The next experiment evaluates the effect of balancing class instances to address the class imbalance problem. Balancing class instances is achieved by performing random under-sampling on the original training dataset (DB) to create a single balanced subset of the training data to be utilized for training. The resulting balanced subset now has an equal number of positive *reason* and negative *non-reason* class instances (Fig.6 (c)).

In this experiment, the recall is much higher than precision for all base learners, SVM (P:0.17/ R:0.84), LR (P:0.19/ R:0.84) and DT (P:0.16/ R:0.83). In fact, the precision is rather low. This indicates that most of the tokens were predicted as belonging to the *reason* class, when in actuality a majority of them belongs to the *non-reason* class. This also explains

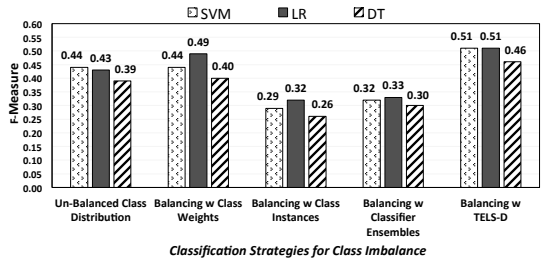


Figure 7: Comparison of classification strategies for class imbalance - F-measures of base classifiers

the very high recall, where most of the ground truth labels were also included in the total predictions. This can be explained by the fact that during under-sampling only a random subset of negative class *non-reason* instances were included in the balanced subset. Hence we discarded many potentially useful instances that are important for learning the *reason* class. In this scenario, the base learners cannot learn the predominant characteristics of the negative class well and hence tend to mis-classify those instances more often.

2.4.5 Balancing with Classifier Ensembles

This experiment evaluates the effect of balancing with ensemble of homogeneous classifiers. Balancing with Ensemble of Homogeneous Classifiers is achieved by performing Under-Bagging strategy on the original training dataset (DB) to create multiple under-sampled subsets of the training data (Sec. 2.3.11). Then we train each base learner on all of these subsets. Lastly, we combine them with Majority Voting (Fig.6 (d)).

In this experiment, the recall is much higher than the precision for all base learners, SVM (P:0.20/ R:0.83), LR (P:0.21/ R:0.81) and DT (P:0.18/ R:0.81). These results are similar to the experimental results of Balancing with Class Instances (Sec.2.4.4). Although, both approaches are similar in the creation of a balanced subset, this current approach uses multiple balanced subsets to counter the limitations of using a single balanced subset (i.e. eliminating potentially important negative class instances). However, the Under-Bagging approach uses majority voting to aggregate the results obtained from training the base classifiers on these subsets. We see (Fig.6 (c)) that the precision on a single subset is very low. So even if we take a majority vote of N such classifiers whose individual base results

are erroneous, the final prediction tends to be also incorrect.

2.4.6 Balancing with TELS-D

Our proposed approach, TELS-D is a multi-layer framework (Sec. 2.3.12). The first layer in TELS-D creates a balanced learning environment to handle class imbalance in the training dataset.

This experiment evaluates the first layer in TELS-D. Balancing is achieved by creating multiple balanced subsets (BB) of the original training data (DB) based on the imbalanced level (IM) in the training set. We train each base learner on the balanced subsets (BB) and combine them with Stacking, using another meta-algorithm (Logistic Regression). In contrast to Under-Bagging which uses simple majority voting, TELS-D employs stacking method to combine the results from the base learners and make the final predictions (Fig.6 (e)).

In this experiment, the recall is a little higher than precision for all base learners, SVM (P:0.45/ R:0.60), LR (P:0.44/ R:0.60) and, DT (P:0.39/ R:0.55). That is, although we have predicted many of the tokens correctly, some of the class predictions were incorrect when compared against ground truth labels. This small learning bias towards the minority *reason* class is expected because, during the training phase, we give priority to learning the minority *reason* class well by training on multiple subsets that have the same minority instances.

2.4.7 Comparing Classification Strategies for Class Imbalance

To compare our experimental results of different approaches for dealing with class imbalance, we evaluate their performances on each individual base learners using the F-Measure metric. F-Measure gives a weighted average of the precision and recall scores. An improvement is the F-measure indicates an equilibrium point where we increase the number of correct class predictions thereby decreasing the number of incorrect class predictions. Fig. 7 shows that our proposed **TELS-D** approach is effective in solving the class imbalance

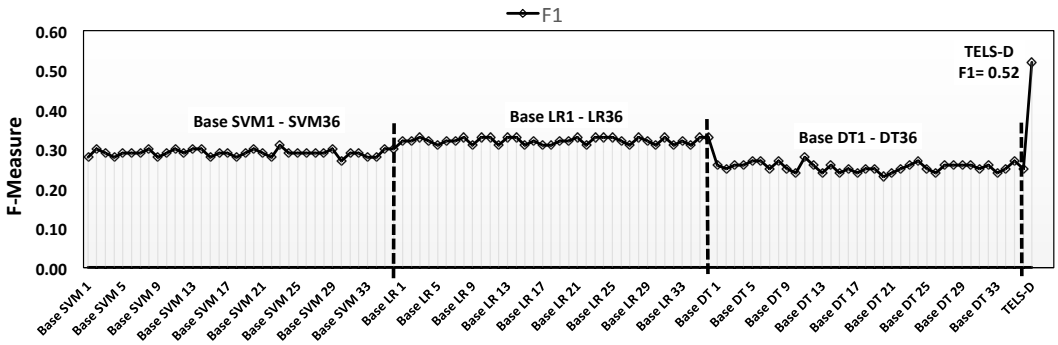


Figure 8: Evaluation of tiered ensemble learning system with diversity (TELS-D)

problem with higher F-Measures on all three base learners (SVM_F-Measure:0.51/ LR_F-Measure:0.51/ DT_F-Measure:0.46) compared to other approaches that deal with class imbalance.

2.4.8 Ensemble Learning with TELS-D

The second layer in TELS-D is designed to create and combine an ensemble of heterogeneous classifiers to improve the accuracy over the individual base learners (Sec. 2.3.12). This experiment evaluates the second layer of TELS-D built on the output from the first layer. The predictions of the three base learners trained over all balanced subsets in the first layer are combined with Stacking using a meta-algorithm, in our case a simple linear algorithm like Logistic Regression (LR), in the second layer.

Fig. 8 shows F-Measure of: 1) Individual base classifiers generated by training the three base learners on all balanced subsets, 2) Ensemble combined with majority voting (for comparison only) and, 3) Ensemble combined with Stacking. The F-measures of individual base classifiers were ranging from 0.28-0.33, Ensemble with Majority voting is 0.22 whereas the F-measure of the ensemble with stacking is 0.52. This experiment demonstrates the power of an ensemble learning system with a learning-over-learners combiner called meta-algorithm in the final step. The meta-algorithm learns from the errors generated by the base classifiers to output the correct result. Majority voting on the other hand is under performing due to the fact that, with simple counting of votes, the errors of the

Table 2: FAERS Examples of Reason class labels predicted by TELS-D

Example	Sentence from the FAERS Narrative	True Positive (TP)	False Positive (FP)	False Negative (FN)
1)	The patient was treated with canagliflozin for *type 2 diabetes* and domperidone for *diabetic gastroparesis*	type, diabetes, diabetic, gastroparesis		2
2)	The patient had previously experienced allergy when taking mycins (antibacterials for *systemic use*)	systemic		use
3)	Concurrent conditions **abdominal pain**, **diabetic** paresis.		abdominal, pain, diabetic	

base classifiers only add up and thus make the final result more erroneous.

We have compared our results with an existing study [74] conducted on the same i2b2 test dataset. [74] demonstrated with MedEx only and SVM-based NER including MedEx. The results showed that for recognizing the *reason* entity from the narratives, the rule-based MedEx system achieved a F-measure of 0.43 while the SVM combined with MedEx achieved 0.48. Our results from TELS-D approach show an improvement over both MedEx and SVM including MedEx with the F-measure of 0.52.

2.4.9 Analysis of TELS-D Results on FAERS Reports:

Due to lack of ground truth labels for FAERS reports, we manually reviewed and evaluated the TELS-D results on few of the 16 FAERS reports. An analysis of errors on one of the FAERS narrative is discussed below (See Table 2).

- True Positives: True Positives (TP) are the correctly predicted tokens. In the Table 2, we can observe that for examples 1 and 2 all the tokens labeled as *reason* class have been accurately predicted as true positives by our TELS-D. Most of the ground truth labeled words in these sentences are purely medical text and follow a certain sentence structure.

- False Positives: False positives (FP), i.e., incorrectly predicted as *reason* class, mostly occurred when the token was not associated with a medication. For instance, example 3 shows that although the incorrectly predicted token is all medical text, it was not associated with a medication name in the same sentence. Hence it cannot be an

indication for taking a medication and is not predicted as *reason*. Cases such as these are very difficult to classify and indicate a need for additional features to learn such patterns in the text.

- False Negatives: Our evaluation showed that false negatives, i.e., incorrectly predicted as non-reason class, occurred primarily due to the mixture of medical and non-medical words. Most of the time, we have noticed that these false negative tokens are embedded or were a part of the true positive tokens. For instance, in examples 1 and 2, the words "2", "use" are all commonly used regular text.

2.5 Discussion

Lack of annotated FAERS dataset. First, FAERS narratives cannot be published without data redaction because of privacy concerns. Redaction of these reports requires a huge amount of cautious efforts to make sure no privacy threatening information remains in the publishable text. Since the redaction process requires perfect *recall* with utmost *precision*, it is almost impossible to be accomplished automatically without significant manual intervention. Therefore, creating a large corpus of redacted FAERS narratives is challenging in itself. Second, annotating FAERS narrative requires deep domain knowledge and reviewing experiences. Deployable supervised machine learning models used for such task must be trained on larger datasets annotated by FDA's own safety reviewers whose annotating strategy reflects the reviewing guidelines. However, due to limited resources, annotating a large set of FAERS narratives is not trivial as it requires extra effort and time in addition to the routine drug review tasks. Given the above challenges, there are no publishable FAERS reports annotated by FDA that can be used in this study for training and testing purposes. Therefore, to prove the concept and for the reproducibility of this study, we trained our model and evaluated our methodology using the public benchmark dataset (i2b2 2009 discharge summaries). In addition, we tested the trained model on a few redacted FAERS narratives that have been annotated. Since discharge summaries do

not necessarily share the same vocabulary as the FAERS narratives, we expect this switch in data sets to be reflected in the results as well.

Practical application of this study for FDA. Automatically identifying high value information from the biomedical text has been recognized by FDA as one of the important steps in its regulatory and supervisory tasks. FDA has been partnering with research institutes and technology companies to develop text mining and natural language processing tools for various types of biomedical text collected by FDA such as vaccine ADR reports (VAERS), FAERS reports, and others. Due to the different nature of these texts, the tools and methodologies are highly customized to work with a particular text type. Moreover, among these text types, FAERS narratives have relatively complex structure in terms of size, vocabulary and style of writing. To cope with this complexity, we propose a machine learning framework that can combine some of these internally available existing tools to extract information from FAERS narratives in an ensemble fashion. These extracted results can be further utilized by advanced data mining or visualization techniques to enhance the drug review process.

2.6 Summary

This research work describes a novel approach called Tiered Ensemble Learning System with Diversity (TELS-D) for biomedical NER from Adverse Event Reports. Our proposed approach uses an ensemble of diverse heterogeneous classification methods to recognize named entities in the text while also dealing with the critical problem of skewed class distribution of the named entities in the training datasets. Our results are promising and indicate that, in the context of binary classification an ensemble approach would be a better choice for NER especially for class imbalanced datasets.

3 Sequence labeling for word-level information detection

In this chapter, we propose a three-layer deep learning architecture for the NER subtask, consisting of a BiLSTM layer for character-level encoding, a BiLSTM layer for word-level encoding, and a CRF layer for structured prediction. Our system was ranked first for NER task in the MADE1.0, NLP Challenge for Detecting Medication and Adverse Drug Events from Electronic Health Records hosted by University of Massachusetts at Lowell, Worcester, and Amherst. This work was published in Drug Safety, The Official journal of the International Society of Pharmacovigilance [39]

3.1 Introduction

Drug-related adverse events (ADEs) are known to be a leading cause of death in the United States [2]. Early detection of the ADE incidents aids in the timely assessment, mitigation and prevention of future occurrences of severe, potentially fatal ADEs. Natural Language Processing (NLP) techniques towards recognizing ADEs and related information from spontaneous reports, clinical reports, electronic health records (EHR) provides an effective way of drug safety monitoring and pharmacovigilance.

A major challenge with processing EHR records is that EHR notes, while containing valuable knowledge correspond to unstructured text. Numerous challenges arise when extracting entities from such narratives. Often the notes contain medical and non-medical abbreviations, acronyms, numbers and misspelled words which make it difficult to recognize the critical information in the notes. In other words, certain types of information such as ADEs, indications, signs & symptoms are harder to detect than others such as drug names. This can be explained by the following. First, these entities can span across multiple words, about one to seven words per entity. Also, some entities could be expressed as a combination of entity-specific medical terms as well as non-medical descriptive text [35]. For instance, in the phrase “coronary artery disease related event prophylaxis”, the words “related” and “event” are descriptive text while the rest are medical terms. Moreover,

Table 3: Examples showing key challenges of biomedical text.

Challenges	Example text
Multiple words	*Lymphoplasmacytoid lymphoma involving bone marrow and spleen*
Medical and non-medical words	*cervix again is significantly stenotic*
Abbreviations	*IgG kappa monoclonal protein*
Ambiguous Named Entities	*Headaches - Indication or ADE or Sign or Symptom*

there is a lot of ambiguity among relevant named entities. Depending upon the context, the same exact phrase can be either an ADE, indication or a sign & symptom (SSLIF). Table 3 states key challenges of textual notes. The example text is taken from a de-identified dataset of EHR notes of 21 cancer patients from the University of Massachusetts Medical School.

To tackle the above challenges, an ADE detection system should: 1) capture both syntactic and semantic features of the words to best distinguish between ADE related terms and non-medical words, 2) model the dependencies among words within a sentence so that ADR related phrases consisting of multiple strongly associated words (including non-medical words) can be identified, and 3) master plan the ADE detection by considering the possibly labeling outcomes for each word so that the detected ADE words or phrases as a whole in a sentence make sense. Following this principle, we propose our Dual-Level Embedding for Adverse Drug Event Detection framework, DLADE in short – a three layered deep learning architecture that solves the above three challenges jointly within one model. In addition, due to the noisy nature of the EHR text data, we design a rule-based EHR text preprocessor for providing clean tokenized text input essential for the success of the subsequently applied computational detection method.

3.2 Related Work

Rule-based Method. Rule-based extraction techniques are user-created pattern matching rules that require human expertise. In [95, 28] a rule-based approach that combines rules with semantic lexicons is used to extract drugs and related information such as dosage,

duration and signs or symptoms from clinical records.

Machine Learning Method. In [95, 32] statistical and machine learning techniques such as Hidden Markov Models (HMM) and Conditional Random Fields (CRF) are used to extract information from biomedical text. [33] developed a machine learning based biomedical named entity tagger using Support Vector Machines (SVM), to extract medication and ADE information from medical narratives.

In recent years, deep learning models especially Recurrent Neural Network (RNN) models have been shown to be promising techniques for sequence tagging and Named Entity Recognition (NER) tasks due to their ability to learn from the context surrounding the words in a sequence [58]. Long Short-Term Memory (LSTM) [59] is a type of RNN that is effective at learning the long-term dependencies between words in a sequence. CRFs [96] are probabilistic graphical models that have been used for sequence labeling tasks due to their ability to model the dependencies in the outputs of a sequence. A combination of RNN and CRF models have also been explored and found to be effective for sequence tagging [36, 97, 98]. Most of the deep learning models developed for NER task use word embeddings as input to the models. Word embeddings are vector representations of words in the text. These word embeddings can either be trained on domain specific text such as biomedical texts, EHR notes, PubMed articles [99, 100], or they can be trained on wide variety of general text such as Wikipedia articles [101].

3.3 The MADE1.0 NLP Challenge

This section provides a brief introduction to the MADE1.0 NLP Challenges for Detecting Medication and Adverse Drug Events from Electronic Health Records hosted by University of Massachusetts at Lowell, Worcester, and Amherst . The main objective of the challenge is to advance ADE detection techniques to improve patient safety and health care quality. The challenge consists of the following three tasks: 1) Named entity recognition (NER), 2) Relation identification (RI) and, 3) Integrated task (IT).

3.3.1 The Task

We have developed our system, DLADE (Dual-Level Embedding for Adverse Drug Event Detection) [57], specifically for Task 1, the Named Entity Recognition (NER) problem of the challenge. The task is to develop a system capable of automatically detecting any mentions of medication names and their attributes (dosage, frequency, route, duration) as well as mentions of ADEs, indications, other signs & symptoms. Tasks 2 & 3 (RI & IT) are beyond the scope of this paper.

3.3.2 Data Set

The MADE1.0 challenge used a total of 1089 de-identified EHR notes from 21 cancer patients. The notes are annotated with medication information (such as medication name, dosage, route, frequency, duration), adverse drug events (ADEs), indications and other signs and symptoms. The annotated notes were released in the BioC format [102]. 876 of these reports were released to participants of the competition for developing their learning system along with the gold standard annotation.

3.3.3 Resources

This challenge restricted the usage of existing NLP tools such as NLTK [103], Stanford NLP [104], cTakes [67] which should only be used for text preprocessing, in order to assure fairness among competition participants who included both university as well as company contributors with diverse resource access. The term standard resources refer to the training data released to the participating teams, the pre-trained word embedding trained using wiki, and de-identified Pittsburgh EHR and PubMed articles [73, 36] and Unified Medical Language System (UMLS) [65]. The term extended resources refers to publicly available tools designed to work with medical concepts and medical relations as well as any ancillary corpus in addition to the standard resources. Our system, DLADE, is developed using only the standard resources released as part of the challenge – the training data

and the pre-trained word embedding.

3.3.4 Evaluation Process for MADE1.0 Challenge

The developed system was then evaluated by the MADE1.0 organizers on two different tracks: 1) Standard track using only the standard MADE1.0 resources and, 2) Extended track using customized resources available publicly. The top teams for the AMIA 2018 Informatics Summit panel presentation were selected based only on the performance of each team for the Standard track. The evaluation is based on the strict matching in F1-score using exact phrase-level evaluation. Relaxed matching using word-level evaluation is not considered. The metrics used for evaluating the systems are Precision, Recall, F1-score and the Micro-averaged score which sums up the individual true positives, false positives, and false negatives of the system for different sets and the apply them to get the statistics. The best score is determined by the Micro-averaged F1-score for the Standard track using an exact phrase-level evaluation. This simplified way selected a winner for this task of the competition.

3.4 Methodology

3.4.1 Preprocessing

As we will explain in Section 3.5.2, our model considers the EHR notes as a set of sentences where each individual sentence in turn consists of a sequence of words. Therefore, we first tokenize the EHR notes into sentences and then tokenize the words within each sentence. MADE1.0 EHR notes contain noise (Figure 9), e.g., section headings with repeating punctuations and abnormal text formatting, e.g., unexpected line breaks where existing off the-shelf tokenizers such as NLTK [80] fail to produce promising results which we show in the result section. For this reason, we instead built a rule-based tokenizer that processes the EHR note character by character for sentence and word chunking while concurrently recording the character offsets w.r.t the original text file. The rule-based process the text narrative character by character. It decides the sentence boundary by considering

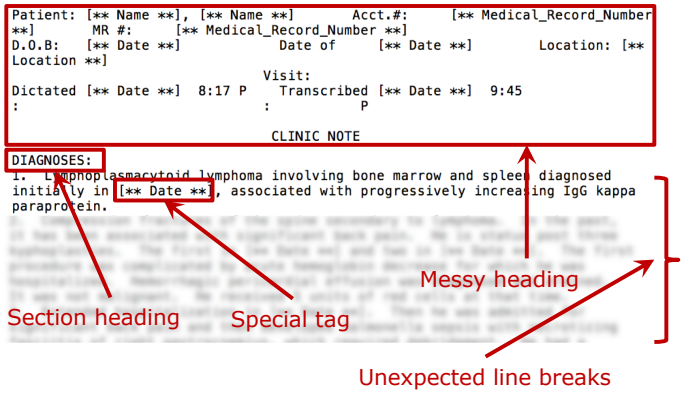


Figure 9: Noise in the EHR text.

Sentence Tokenizer (*Document*):
  For *Character* in *Document*:
    If *Character* is '.' Then:
      If *Previous Word's length* > k and *Previous and Next Character* is not *a Number:*
        Break *a Sentence*
    Else:
      Continue

Figure 10: Algorithm 1 Rule-based Sentence Tokenizer.

the period sign with additional conditions to avoid false alarm such as "Dr.", "Mr.", "1.23" etc. The sentences (except the first one which usually consists of encrypted headings) are then passed to a rule-based word tokenizer which decides the word boundary by considering the spacing among the words. The tokenizers also record the boundary offsets for evaluation purpose. The source code of the preprocessor is released.

Some named entities correspond to multiple words. Hence we use the IOB [105] tagging scheme to distinguish between the beginning of an entity (tag B-named entity), or the inside of an entity (tag I-named entity). The no-entity tag is O.

3.4.2 Word Embedding

Word embeddings are dense representations of words that encode both syntactic and semantic features of the words into a vector. Each word is mapped to a real-valued vector of a low dimensional space, the dimensionality typically ranging between 50 to few hundreds (such as 200, 300). This is much smaller compared to the one-hot vector representations of

the words whose dimension is usually in thousands with sparse vectors.

Word embeddings have shown to improve the performance of sequence tagging tasks [106] and are an integral part of the deep learning models. Word embeddings can be learned from the training data if the training corpus is large enough and has a good vocabulary size. In this case, the vectors are randomly initialized and passed to the neural network in order to learn and further tune the random vectors to provide a good meaningful representation of the words. Alternatively, there are publicly available pre-trained word embeddings which can be readily passed as input to the deep learning models. Some of these pre-trained word embeddings are trained specifically for a domain or task such as biomedical text [99, 100] while others are more general purpose and are trained on Wikipedia articles [101]. The pre-trained embeddings can either be fixed while training the network or can be further tuned to make them better representations specific to the task.

Word embeddings can also be learned from the characters in the word. Character-level representations of the word capture the morphological features such the prefix or the suffix of a word, words starting with upper case letters and abbreviations which can be encoded into a dense representation [107]. The character-level embeddings can be used to supplement the learned or pre-trained word-level embeddings to train a deep learning model.

In the context of ADE detection task, one of the challenges with EHR text (as shown in Table 3) is that it comprises of various medical terms and abbreviations. The pre-trained word embedding for such words and phrases might not be available especially if they occur not too frequently in the corpus on which the word embeddings are learned from. In such cases where a pre-trained word embedding is unavailable, the learned character level representation of the word will enable us to extract the meaning of words. Neural network models such as Convolutional Neural Networks (CNN) or RNNs can be used to run over the sequence of characters in a word to learn the character-level representation of the word.

3.4.3 Methods

In this section, we describe the methods used in our system: Bi-LSTM, CRF, and Bi-LSTM-CRF.

Bi-LSTM. Recurrent Neural Network (RNN) models are designed to capture the long-term dependencies in a sequence. They have an input layer, hidden layer and output layer. The input layer takes the word features in the form of word embeddings. The hidden layer maintains information on previous outputs enabling it to predict the current output based on the past information and previous word in the sequence. The output layer produces the probability distribution for each label. However, RNNs are less effective with longer sequences due to the problem with vanishing or exploding gradient [108, 109] and thus result in a network that cannot learn well from the longer training sequences.

Long Short-Term Memory (LSTM) neural network [32] is a type of RNN that are designed to overcome the gradient vanishing/exploding problem and thus efficiently learning the long-term dependencies in a sequence [110]. They have a built-in memory cell within the hidden layer which is responsible for controlling the flow of previous outputs to the current output without exploding the gradient. However, a LSTM network only captures information about the previous context and does not take into account the future context of the current output.

Bi-LSTM networks [111] have proven to be very useful to capture the entire context by processing the sequence in both forward and backward directions with two hidden layers, one for each direction. The output from both directions is concatenated to form the final output. In the context of ADE detection task, the sentences in the EHRs are often long sequences comprising of named entities that often span across multiple words within the sequence. The named entities are also heavily dependent upon the context they occur in and more often the same word or phrase can be tagged as two different named entities depending upon the context.

CRF. CRF models [96] are widely used for sequence labeling tasks. Given a sequence,

the model uses contextual information from preceding and succeeding information in the sequence to predict the current label. The models predict the label sequence jointly instead of predicting each label individually. These models can predict sequences where multiple words depend on each other. In the context of ADE detection, one of the challenges with EHR text is that the named entities can occur as a combination of medical and non-medical words. For instance, in the named entity phrase "cervix again is significantly stenotic" the label for each of the words in the phrase is greatly dependent on the label of the previous word.

Bi-LSTM and CRF. LSTM and CRF have their own advantages and disadvantages. LSTM is better for modeling long sequences of words, but the label for each word is predicted independently and not as a part of the sequence. CRF is better for modeling the entire sequence jointly, but need hand crafted features to obtain significantly good results. A combination of Bi-LSTM and CRF models [58] have been used for sequence tagging where each one of the models contributes to the combine model while complementing each other. In the sequence tagging task, Bi-LSTM is used to capture the contextual representation of the words from the input features. The output from the Bi-LSTM are fed to the CRF layer to jointly predict the best label sequence.

DLADE Model. Given the success of deep learning models for NLP tasks [36, 112], we have developed a deep learning based system that utilizes the combined effectiveness of RNNs, more precisely Bi-directional long short-term memory (Bi-LSTM) [111] models and CRF by integrating them into one deep network architecture. The Bi-LSTM networks have been widely used for NLP tasks to learn the context representation of a word in a sequence by traversing through the sequence in both forward and backward (i.e. reverse order) directions.

In a nutshell, our model is composed of a Bi-LSTM neural network for an input layer responsible for character embedding, a second Bi-LSTM for word embedding followed by a linear-chain CRF output layer. We have used the pre-trained medical word embedding provided by the MADE1.0 challenge [73, 36]. More precisely, first at the bottom,

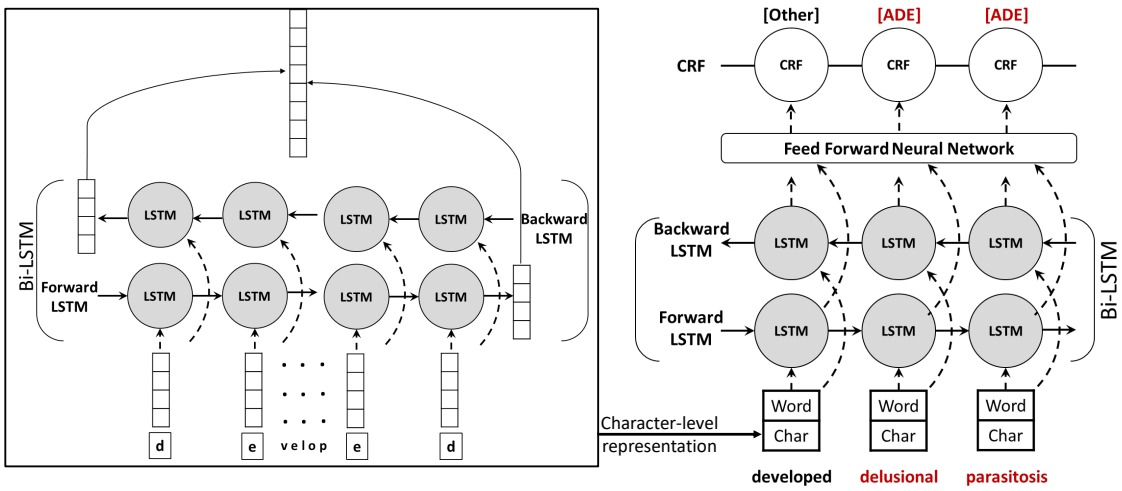


Figure 11: DLADE System Architecture

character-level representations which capture the morphology of a word are computed by running a bidirectional-LSTM over the sequence of characters in the input words. A consolidated dense embedding, comprised of pre-trained medical word embedding concatenated with a learned character-level representation, is used to represent a word. Figure 11 shows our system architecture. Although for the MADE1.0 challenge, we have used the MADE1.0 pre-trained word embedding, our system is designed to plug-and-play with any pre-trained word embedding.

We feed this dense embedding of each word into a second bidirectional-LSTM. This second Bi-LSTM then extracts the contextual representation of each word in the sentence that captures information from the meaning of the word, its characters and its context. The output from the bidirectional-LSTM is used as input to a feed-forward neural network to compute a vector of scores, where each entry corresponds to a score for each tag. Tags are the individual named entities. To make the final prediction, the output of the feed-forward network is passed to a liner-chain CRF. The overall model is trained by minimizing the negative log-likelihood.

Table 4: Evaluation Results on the Final MADE1.0 Holdout Test Set.

	ADE	Dose	Drug	Duration	Frequency	Indication	Route	Severity	SSLIF	Micro-Avg
Precision	0.7261	0.8721	0.9066	0.7143	0.8438	0.6587	0.9100	0.7798	0.8309	0.8373
Recall	0.5644	0.8874	0.9019	0.8271	0.8412	0.6216	0.9381	0.8362	0.8570	0.8454
F1-Score	0.6351	0.8797	0.9042	0.7666	0.8425	0.6396	0.9239	0.8070	0.8438	0.8413

Table 5: Evaluation Results on the Final MADE1.0 Holdout Test Set with NTLK Tokenizer.

	ADE	Dose	Drug	Duration	Frequency	Indication	Route	Severity	SSLIF	Micro-Avg
Precision	0.7081	0.8451	0.9066	0.8889	0.8398	0.7024	0.9073	0.7649	0.8206	0.8337
Recall	0.5236	0.8811	0.8988	0.782	0.8563	0.6006	0.9330	0.8723	0.8654	0.8474
F1-Score	0.6020	0.8627	0.9027	0.8320	0.8480	0.6475	0.9200	0.8151	0.8424	0.8405

3.5 Experimental Results

3.5.1 Hyperparameter Settings

The named entities are Drug, Indication, Frequency, Severity, Dose, Duration, Route, ADE, SSLIF (other sign, symptom or disease). The model operates on the tokenized sentences. We used a batch size of 20 sentences. We did not make any restrictions on the sentence length. Rather, we used the maximum length of the sentences in a batch. All shorter sentences in that batch are padded with masks. As input, the pre-trained word embedding are 200 dimensional vectors and the learned character-level embedding are 100 dimensional vectors. The hidden state is set to 100 dimensions for running Bi-LSTM for learning character embedding. The hidden state is set to 300 dimensions for running Bi-LSTM with dense word embedding. To avoid overfitting, we apply a dropout strategy [112, 113] of 0.5 for our model. All the models were trained with learning rate of 0.001 using Adam [114]. Our models are trained on Intel(R) Xeon(R) 2.10GHz with a total memory of 251GB. They are implemented using the Tensorflow framework [115].

3.5.2 Methodology

Our system DLADE is trained on the 876 EHR notes from MADE1.0. From the training set of sentences, 10% of the sentences are held out as validation set. This allows us to evaluate the model in the training phase by determining the best F1-score for early stopping. If there is no improvement in the F1-score within the last three consecutive epochs, the system

performs an early stopping.

3.5.3 Results on MADE1.0 Test Data Set

On the evaluation test set consisting of 213 EHR notes, our deep network achieves a micro-averaged Precision, Recall and F1-score of 0.8373, 0.8454, and 0.8413, respectively for the exact phrase-level evaluation. Table 4 shows our evaluation results on the MADE1.0 evaluation test set for each of the entities. Our system has been selected as one of the top three performers and, is ranked first in the MADE1.0 challenge for the Standard NER task.

To demonstrate the effectiveness of our rule-based tokenizer, we compare the prediction results from DLADE which uses our proposed rule-based tokenizer with a baseline system that uses NLTK tokenizer. Table 5 shows that the baseline system achieves higher Micro-averaged recall however gets lower Micro-averaged F1-score. To demonstrate the effectiveness of utilizing dual-level embedding, we compare the prediction results from DLADE, which uses both the learned character-level representations of a word and the pre-trained word-level embedding with a baseline system that utilizes only the pre-trained word-level embedding.

Table 6 compares the F1-scores of individual entities as well as the overall micro-averaged F1-score of all entities combined. It shows the percentage improvement with DLADE using dual-level embedding over the baseline system using only word embedding. We use the pairwise t-test to examine the statistical significance of the differences in performance scores obtained from the two systems on the same test set. F1-scores of individual named entity types as well as the overall (micro-averaged) from both systems are paired. The improvement in F1-score for DLADE as compared to our baseline is statistically significant (p-value <0.05 and p-value <0.01). Of all the entities, Duration showed a large improvement (11.4%) from utilizing the dual-level embedding. Duration labels are challenging to detect because they often are comprised of phrases that contain non-medical text and contain numbers such as, “four cycles”, “14 days”, “day 1 through 14”, “over 15 minutes”, “two weeks”. They can be easily misclassified and treated as the Outside or

Table 6: Improvement for MADE1.0 in F1-score when using Dual-Level Embedding.

	Word Embedding	Dual-Level (Character + Word)Embedding	Improvement
ADE	0.5848	0.6351	8.6%
Dose	0.8172	0.8797	7.6%
Drug	0.8780	0.9042	3.0%
Duration	0.6879	0.7666	11.4%
Frequency	0.7964	0.8425	5.8%
Indication	0.6151	0.6396	4.0%
Route	0.8705	0.9239	6.1%
Severity	0.7648	0.8070	5.5%
SSLIF	0.8290	0.8438	1.8%
Micro-averaged	0.8147	0.8413	3.3%

Table 7: Percentage change in F1-scores.

	F1-score with using word embedding			Percentage change in F1-scores		
	1. MADE1.0	2. GloVE	3. PubMed	4. MADE1.0 over GloVe (%)	5. MADE1.0 over PubMed (%)	6. GloVe over PubMed (%)
ADE	0.6351	0.6197	0.6055	2.48%	4.88%	2.34%
Dose	0.8797	0.8787	0.8575	0.11%	2.58%	2.47%
Drug	0.9042	0.9100	0.8838	-0.63%	2.31%	2.96%
Duration	0.7666	0.8015	0.7943	-4.36%	-3.50%	0.90%
Frequency	0.8425	0.8529	0.8580	-1.22%	-1.81%	-0.59%
Indication	0.6396	0.6512	0.6429	-1.78%	-0.52%	1.28%
Route	0.9239	0.9133	0.9221	1.16%	0.19%	-0.96%
Severity	0.8070	0.8209	0.8098	-1.70%	-0.35%	1.37%
SSLIF	0.8438	0.8453	0.8454	-0.18%	-0.19%	-0.01%
Micro-averaged	0.8413	0.8451	0.8372	-0.45%	0.49%	0.94%

no-entity tag O.

3.5.4 Impact of Pre-trained Word Embedding

We demonstrate the effect of using different pre-trained word embedding in the input layer, we compare the results from DLADE, which uses domain and task specific MADE1.0 word embedding trained using wiki, and Pittsburgh EHR and PubMed articles (1,352,550 word vectors) [36, 73], with two systems that use: 1) general purpose GloVe Common Crawl 840B, 300 dimensional word embedding [101] (4,087,447 word vectors) and, 2) the domain-specific PubMed, 200 dimensional word embedding induced from a combination of PubMed and PMC texts using the word2vec tool for biomedical data purposes [116] (2,196,016 word vectors).

Table 7 shows the F1-scores with using different pre-trained word embedding (columns: 1, 2, 3) and the percentage change in F1-scores with each type of word embedding over the

others (columns: 4, 5, 6). We use the pairwise t-test on the F1-score of the individual entity types as well as the overall (micro-averaged) to determine if these differences are statistically significant. Our results indicate that for detecting some of the entity types there is a minor improvement in the F1-scores with using MADE1.0 word embedding over GloVe (ADE, Dose, Route) or PubMed (ADE, Dose, Drug, Route). However, these improvements are not statistically significant (p p-value $>$0.05). Although, there is a 0.49% improvement in the overall micro-averaged F1-score with using MADE1.0 over PubMed, it is not statistically significant (p $>$0.05). However, the 0.94% improvement with using GloVe over PubMed is statistically significant (p $<$0.05) for this EHR dataset and ADE detection task.

3.5.5 Error Analysis of DLADE System

An error analysis was performed to understand the source of errors generated by the NER system. We inspected and evaluated instances for which the system incorrectly predicted the phrases, considering both false positive and false negative cases.

- One of the challenges as shown in Table 3 is that the entity can span across multiple words. In this case, it is critical to extract the phrase in its entirety to retain the true meaning of the phrase. For this example, our system was able to correctly extract the entire phrase "nodular sclerosing Hodgkin disease involving the mediastinum and both necks". This contains 10 words. However, the phrase was misclassified as Indication when it actually is an SSLIF.

- Another challenge is the mixture of medical and non-medical text in the entity phrase. This makes it difficult to detect the entity as a whole. For instance, the phrase "inflammation of your liver or gallbladder or your pancreas" was annotated as SSLIF. Although our system detected the phrase correctly as SSLIF, it missed the last two words "your pancreas" of the phrase. This meant that our result was labeled as Other entity-O wrongly even though it mostly was correct.

- The occurrence of medical abbreviations text is rare in the training set. Although our

system was able to correctly detect certain entities that contain abbreviations such as "stage IIA" (Severity), "HPV" (SSLIF), there are few other entities with abbreviations such as "SIL cytology" (SSLIF) where our system failed to recognize the phrase and categorized it as a no-entity label O.

- Due to the ambiguous nature of Indication, ADE, and SSLIF entity words and phrases, it is very challenging to differentiate between these two types of labels. For example, in the two sentences: 1) "the back pain (Indication) started about 10 o'clock last night" and, 2) "reports weight gain (ADE) and increased (ADE) appetite from corticosteroid therapy", our system misclassified the Indication and ADE labels as SSLIF.

3.6 Discussion

In this paper, we report our experience with MADE1.0 competition and describe our system which is ranked first in the NER task. We study the problem of detecting ADE related terms and phrases from the EHRs. Unlike other research domain such as computer vision where billions of labeled images are made publicly available for research purpose, making large scale labeled medical corpus publicly available is an open challenge as it is a human resource intensive task and it involves many legal issues. We appreciate the effort of the MADE organizers who provides annotated EHRs. However, the size of the corpus is still considerably small. The generality of our method could not be validated even with the great results we have shown in the experiments.

For this competition, we use the same methodology for all entity types. However, the challenge of detecting each individual entity type varies. For example, the ability to capture morphological features is important to the entity types if they often consist of special representation as compared to common words whereas the ability to capture the context information is crucial to differentiate "Indication" and "ADE" as they may share the same vocabulary but are expressed differently in the text narrative.

Overall, our system achieved excellent detection accuracy with a micro-averaged Preci-

sion: 0.8373, Recall: 0.8454 and, F1-score: 0.8413. However, the detection accuracy among the nine individual entity types varied with, some entity types achieving better F1-scores such as Route (0.92), Drug (0.90), Dose (0.88), SSLIF (0.84), Frequency (0.84), Severity (0.81), Duration (0.77) over other entity types ADE (0.64) and Indication (0.64). Given that ADE and related information detection from EHR is a challenging task, our system showed an incremental improvement in the scores compared to the benchmark studies [36, 73]. Yet ADE and Indication have proved to be the most challenging of the entity types to detect with a lower recall (0.56 and 0.62 respectively). These challenging entity types might require customized models that are able to tackle the issues with ambiguity that is often encountered while detecting these entities.

3.7 Summary

We have shown that the integration of two widely used sequence labeling techniques that complement each other along with dual-level embedding (character-level and word-level) to represent words in the input layer results in a deep learning architecture that achieves excellent information extraction accuracy for EHR notes. Our system was ranked first in MADE1.0 competition for the NER task. Additional work must be done to improve the accuracy in detecting the challenging entity types such as ADE and Indication. In the future, we will further analyze the results for these entity type and design customized models to improve the detection performance of each individual entity type as well as the overall performance for all entity types.

4 Multi-grained joint model for word and sentence-level information detection

In this chapter, we propose a dual-attention based multi-grained joint deep network model to concurrently solve both ADE entity recognition (fine-grained) and ADE sentence classification (coarse-grained) tasks. Our model improves state-of-art F1-score for both tasks: (i) entity recognition of ADE words (12.5% increase) and (ii) ADE sentence classification (13.6% increase) on MADE 1.0 benchmark of EHR notes. This work was published in [50]

4.1 Introduction

Background. Adverse drug events (ADEs), injuries resulting from medical intervention, are a leading cause of death in the United States and cost around $30~$130 billion every year [2]. Early detection of ADE incidents aids in the timely assessment, mitigation and prevention of future occurrences of ADEs. Natural Language Processing techniques have been recognized as instrumental in identifying ADEs and related information from unstructured text fields of spontaneous reports and electronic health records (EHRs) and thus in improving drug safety monitoring and pharmacovigilance [78].

Fine-grained ADE detection identifies named ADE entities at the word-level, while *coarse-grained ADE detection* (also ADE assertive text classification) identifies complete sentences describing drug-related adverse effects. [43]'s system for identification of ADE assertive sentences in medical case reports targets the important application of detecting under-reported and under-documented adverse drug effects. Lastly, *multi-grained ADE detection* identifies ADE information at multiple levels of granularity, namely, both entity and sentence level.

As example, Figure 12 displays ADE and non-ADE sentences. The first is an ADE sentence where the mentions of Drugname and ADE entities have the appropriate relationship with each other. Second and third sentences show that the mention of an ADE entity by itself is not sufficient to assert a drug-related adverse side effect.

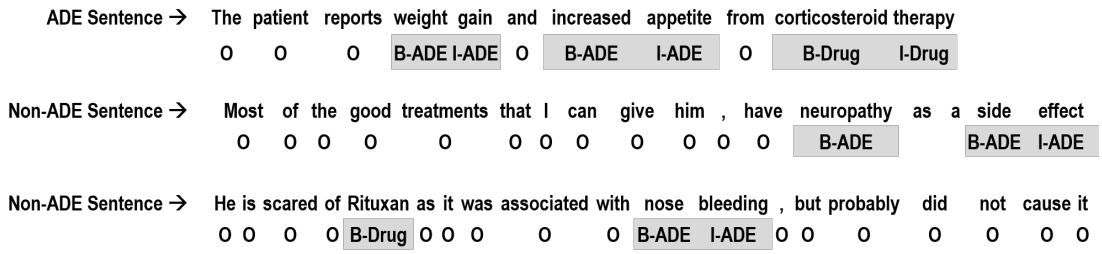


Figure 12: Each sentence is classified as ADE sentence (binary yes/no). Each word is labeled using beginning of an entity (B-...) vs inside an entity (I-...) for ADE related named entities (multiple classes). O denotes no entity tag.

Recently, deep learning-based sequence approaches have shown some promise in extracting fine-grained ADEs and related named entities from text [52]. However, the prevalence of entity-type ambiguity remains a major hurdle, such as, distinguishing between *Indication entities* as the reason for taking a drug versus *ADE entities* as unintended outcomes of taking a drug. Coarse-grained sentence-level detection performs well in identifying ADE descriptive sentences, but is not equipped to detect fine-grained information such as words associated with ADE related named entities. Unfortunately, when the interaction between these two extraction tasks is ignored, we miss the opportunity of the transfer of knowledge between the ADE entity and sentence prediction tasks.

Attention-based neural network models have been shown to be effective for text classification tasks [117, 17] from alignment attention in translation [118] to supervising attention in binary text classification [119]. Previous approaches typically apply only a single round of attention focusing on simple semantic information In our ADE detection task, instead, key elements of the sentence can be linked to multiple categories of task-specific semantic information of the named entities (ADE, Drug, Indication, Severity, Dose etc.). Thus, single attention is insufficient in exploring this multi-aspect information and consequently risks losing important cues.

Proposed Approach. In our work, we tackle the above shortcomings by designing a dual-attention based neural network model for multi-grained joint learning, called MGADE, that jointly identifies both ADE entities and ADE assertive sentences. The design of MGADE is inspired by multi-task Recurrent Neural Network architectures for jointly learning to la-

bel tokens and sentences in a binary classification setting [119]. In addition, our model makes use of a supervised self-attention mechanism based on entity-level predictions to guide the attention function – aiding it in tackling the above entity-type ambiguity problem. We also introduce novel strategies of constructing multiple complementary sentence-level representations to enhance the performance of sentence classification.

Our key contributions include:

1. *Joint Model.* We jointly model ADE entity recognition as a multi-class sequence tagging problem and ADE assertive text classification as binary classification. Our model leverages the mutually beneficial relationships between these two tasks, e.g., ADE sentence classification can influence ADE entity recognition by identifying clues that contribute to ADE assertiveness of the sentence and match them to ADE entities.
2. *Dual-Attention.* Our novel method for generating and pooling multiple attention mechanisms produces informative sentence-level representations. Our dual-attention mechanisms based on word-level entity predictions construct multiple representations of the same sentence. The dual-attention weighted sentence-level representations capture both task-specific and semantic information in a sentence, providing stronger emphasis on key elements essential for sentence classification.
3. *Label-Awareness.* We introduce an augmented sentence-level representation comprised of predicted entity labels which adds label-context to the proposed dual-attention sentence-level representation for better capturing the word-level label distribution and word dependencies within the sentence. This further boosts the performance of the sentence classification task.
4. *Model Evaluation.* We compare our joint model with state-of-art methods for the ADE entity recognition and ADE sentence classification tasks. Experiments on MADE1.0 benchmark of EHR notes [120] demonstrate that our MGADE model drives up the F1-score for both tasks significantly: (i) entity recognition of ADE words by 12.5% and by 23.5% and (ii) ADE sentence classification by 13.6% and by 23.0%, compared to state-of-art single task and joint-task models, respectively.

4.2 Related Work

Fine-grained ADE Detection. Jagannatha and Yu [36] have employed a bidirectional LSTM-CRF model to label named entities from electronic health records of cancer patients. Pandey et al. [19] proposed a bidirectional recurrent neural network with attention to extract ADRs and classify the relationship between entities from Medline abstracts and EHR datasets. Wunnava et al. [39] presented a three-layer deep learning architecture for identifying named entities from EHRs, consisting of a Bi-LSTM layer for character-level encoding, a Bi-LSTM layer for word-level encoding, and a CRF layer for structured prediction.

Coarse-grained ADE Detection. Huynh et al. [46] applies Convolutional Neural Networks using pre-trained word embeddings to detect sentences describing ADEs. Tafti et al. [47] utilized a feed-forward ANN to discover ADE sentences on PubMed Central data and social media. Dev et al. [24] developed a binary document classifier using logistic regression, random forests and LSTMs to classify an AE case as serious vs. non-serious.

Multi-grained ADE Detection. Zhang et al. [49] developed a multi-task learning model that combines entity recognition with document classification to extract the adverse event from a case narrative and classify the case as serious or non-serious. However, they fall short in tackling our problem. Not only do their targeted labels not fall into the drug-related adverse side effects category in which a causal relationship is suspected and required, but their attention model is only simple self-attention. As consequence, MGADE outperforms their model by 23.5% in F1 score for entity recognition and 23.0% for assertive text classification as seen in Section 4.4.

4.3 Methodology

4.3.1 Task Definition

In the ADE and medication related information detection task, the entities are *ADE, Drug-name, Dose, Duration, Frequency, Indication, Route, Severity* and *Other Signs & Symptoms*. The no-entity tag is O. Because some entities (like weight gain) can have multiple words, we

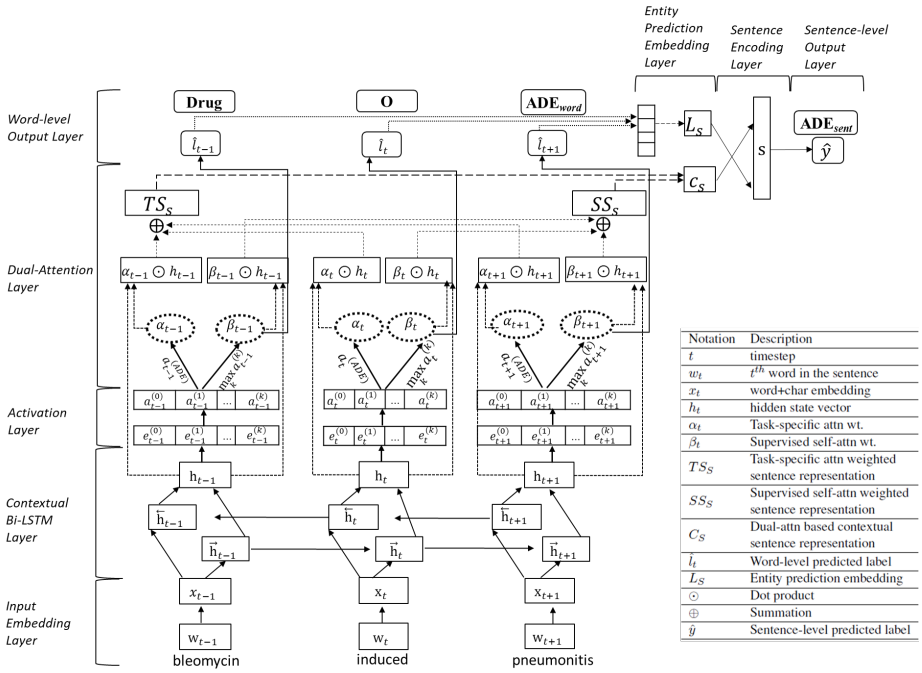


Figure 13: The architecture of the proposed **M**ulti-**G**rained **ADE** Detection Network (MGADE)

work with a BIO tagging scheme to distinguish between beginning (tag B-...) versus inside of an entity (tag I-...). The notation we use is given in Fig 13. Given a sentence (a sequence of words), task one is the multi-class classification of ADE and medication related named entities in the text sequence, i.e., entity recognition. Task two is the binary classification of a sentence as ADE assertive text. The overall goal is to minimize the weighted sum of entity recognition loss and sentence classification loss.

4.3.2 Input Embedding Layer

The input of this layer is a sentence represented by a sequence of words $S = \langle w_1, w_2, ..., w_N \rangle$, where N is sentence length. The words are first broken into individual characters and character-level representations which capture the morphology of a word computed with a bidirectional-LSTM over the sequence of characters in the input words. We employ the pre-trained word vector, GloVe (Pennington et al., 2014), to obtain a fixed word embedding of each word. A consolidated dense embedding, comprised of pre-trained word embedding

concatenated with a learned character-level representation, is used to represent a word. The output of this layer is $X = [x_1, x_2, ..., x_N]$.

4.3.3 Contextual Layer

LSTM is a type of recurrent neural network that effectively captures long-distance sequence information and the interaction between adjacent words [59]. The word representations x_t are given as input to two separate LSTM networks (Bi-LSTM) that scan the sequence forward and backward, respectively. The hidden states learned by the forward and backward LSTMs are denoted as $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$, respectively.

$$\overrightarrow{h}_t = LSTM\left(x_t, \overrightarrow{h}_{t-1}\right) \tag{5}$$

$$\overleftarrow{h}_t = LSTM\left(x_t, \overleftarrow{h}_{t+1}\right) \tag{6}$$

The output of this layer is a sequence of hidden states $H = [h_1, h_2, ..., h_N]$, where h_t is a concatenation of $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$. This way, the hidden state h_t of a word encodes information about the t^{th} word and its context:

$$h_t = \left[\overrightarrow{h}_t; \overleftarrow{h}_t\right] \tag{7}$$

4.3.4 Word-level (NER) Output Layer

The hidden states h_t are passed through a non-linear layer and then with the softmax activation function to k output nodes, where k denotes the number of entity-types (classes). Entity-type labels are the named entities in the BIO format. Each output node belongs to some entity-type and outputs a score for that entity-type. The output of the softmax function is a categorical probability distribution, where output probabilities of each class

is between 0 and 1, and the total sum of all output probabilities is equal to 1.

$$a_t^{(i)} = \frac{\exp\left(e_t^{(i)}\right)}{\sum_{j=1}^{k} \exp\left(e_t^{(j)}\right)} \tag{8}$$

Data is classified into a entity-type that has the highest probability value.

$$\hat{a}_t = \max_{i \in \{1,2,\ldots,k\}} a_t^{(i)} \tag{9}$$

4.3.5 Dual-Attention Layer

The purpose of the attention mechanism in the sentence classification task is to select important words in different contexts to build informative sentence representations. Different words have different importance for ADE sentence classification task. For instance, key elements (words/phrases) in the ADE detection task are linked to multiple aspects of semantic information associated with the named entity categories - *ADE, Drugname, Severity, Dose, Duration, Indication...etc*. It is necessary to assign the weight for each word according to its contribution to the ADE sentence classification task.

Moreover, certain named entities are task-specific and are considered essential for ADE sentence classification. There exists a direct correspondence between such task-specific named entities and the sentence. Hence, we anticipate that there would be at least one word of the same label as the sentence-level label. For instance, a sentence that is labeled as an ADE sentence has a corresponding ADE entity word. Although other named entity words detect important information and contribute to the ADE sentence-level classification task, a stronger focus should be on task-specific ADE words indicative of the ADE sentence core message. A single attention distribution tends to be insufficient to explore the multi-aspect information and consequently may risk losing important cues [121].

We address this challenge by generating and using multiple attention distributions that offer additional opportunities to extract relevant semantic information. This way, we focus on different aspects of an ADE sentence to create a more informative representation. For

this, we introduce a novel dual-attention mechanism, which in addition to selecting the important semantic areas in the sentence (henceforth referred as supervised self-attention [17, 122, 119]), it also provides stronger emphasis on task-specific semantic aspect areas (henceforth referred as task-specific attention). The task-specific attention promotes the words important to the ADE sentence-classification task and reduces the noise introduced by words which are less important for the task.

Similar to [119, 122], we use a self-attention mechanism where, based on softmax probabilities and normalization, attention-weights are extracted from word-level prediction scores. The difference between the two attention mechanism is that the supervised self-attention recognizes word-level prediction scores of all named entities while the task-specific attention recognizes word-level prediction scores w.r.t only selective named entities (one which correspond to the ADE sentence and ignores other named entities). Specifically, the weights of the supervised self-attention and task-specific attention are calculated as follows:

Word-level prediction w.r.t the task-specific named entity (i.e.,) ***ADE***:

$$a_t^{(\boldsymbol{ADE}_{entity})} = \frac{\exp\left(e_t^{(\boldsymbol{ADE}_{entity})}\right)}{\sum_{j=1}^{k} \exp\left(e_t^{(j)}\right)} \tag{10}$$

Task-specific Attention Weight, normalized to sum up to 1 over all values in the sentence, is:

$$\alpha_t = \frac{a_t^{(\boldsymbol{ADE}_{entity})}}{\sum_{n=1}^{N}\left(a_n^{(\boldsymbol{ADE}_{entity})}\right)} \tag{11}$$

Supervised Self-Attention Weight, normalized to sum up to 1 over all values in the sentence:

$$\beta_t = \frac{\hat{a}_t}{\sum_{n=1}^{N} \hat{a}_n} \tag{12}$$

Fig 14 shows the examples of the supervised self-attention and task-specific attention distributions generated from our attention layer. The color depth expresses the degree

(a) Task-specific Attention **(b)** Supervised Self-attention

(c) Distribution of attention weights.

Figure 14: Attention Visualizations: Highlighted words indicate attended words. Stronger color denotes higher focus of attention. (a) Task-specific attention: Recognizes task-specific semantic aspect areas of sentence, with focus on ADE entity words essential for ADE sentence classification task. (b) Supervised Self-attention: Recognizes all important areas in the sentence. (c) Distribution of Task-specific attention and Supervised Self-attention weights.

of importance of the weight in attention vector. As depicted in Fig. 14, the task-specific attention emphasizes more on the parts relevant to the ADE sentence classification task.

Attention-based Sentence Representations. To generate informative and more accurate sentence representations, we construct two different sentence representations as a weighted sum of the context-conditioned hidden states using the task-specific attention weight α_t and supervised self-attention weight β_t, respectively.

1. Task-specific attention weighted sentence rep.:

$$TS_S = \sum_{t=1}^{N} \alpha_t h_t \tag{13}$$

2. Supervised self-attention weighted sentence rep.:

$$SS_S = \sum_{t=1}^{N} \beta_t h_t \tag{14}$$

Attention Pooling A combination of multiple sentence representations obtained from focusing on different aspects captures the overall contextual semantic information about a sentence. The two attention-based representations are concatenated to form a dual-attention contextual sentence representation:

$$C_S = [TS_S; SS_S] \tag{15}$$

4.3.6 Entity Prediction Embedding Layer

ADE detection is a challenging task. Understanding the co-occurrence of named entities (labels) is essential for ADE sentence classification. Although we implicitly capture long-range label dependencies with Bi-LSTM in the contextual layer, and make even more informative sentence-level representations with the help of the dual-attention layer, explicitly integrating information on the label-distribution in a sentence is further helpful to understand the label co-occurrence structure and dependencies in the sentence. The idea is to further improve the performance of ADE sentence classification task by learning the output word-level label knowledge. For a better representing of the word-level label distribution and to capture potential label dependencies within each sentence, we propose Entity Prediction Embedding (EPE), a sentence-level vector representation of entity labels predicted at the word-level output layer (Sec. 4.3.4).

$$\hat{l}_t = \underset{i \in \{0,1,2,...,k\}}{\arg\max}\ a_t^{(i)} \tag{16}$$

$$L_S = [v_0, v_1, v_2, ..., v_k]\,; v_i \in \{0, 1\} \tag{17}$$

4.3.7 Sentence Encoding Layer

A final sentence representation that captures the overall contextual semantic information and label dependencies within the sentence is constructed by combining the dual-attention weighted sentence representation and Entity Prediction Embedding, respectively.

$$\mathbf{S} = [\mathbf{C}_S; \mathbf{L}_S] \tag{18}$$

4.3.8 Sentence Classification Output Layer

Finally, we apply a fully connected function and use sigmoid activation to output the sentence prediction score.

$$\hat{y}^{sentence} = p\left(y^{(j=1)} \mid S\right) \tag{19}$$

4.3.9 Optimization objective

The objective is to minimize the mean squared error between the predicted sentence-level score $\hat{y}^{(sentence)}$ and the gold-standard sentence label $y^{(sentence)}$ across all m sentences:

$$L_{sentence} = \sum_{m} \left(y^{(m)} - \hat{y}^{(m)}\right)^2 \tag{20}$$

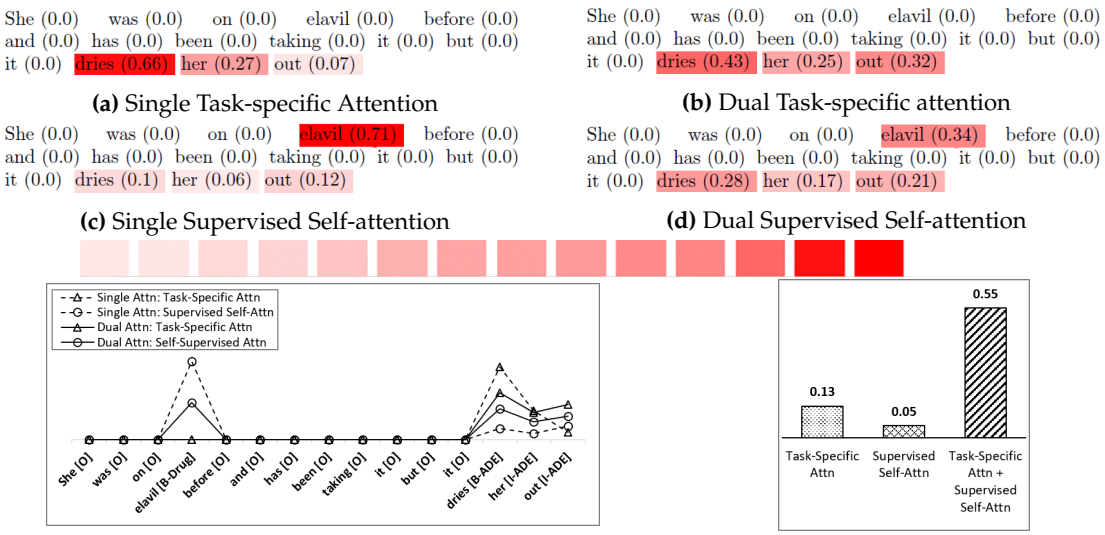
The objective is to minimize the cross-entropy loss between the predicted word-level probability score $\hat{y}^{(entity)}$ and the gold-standard sentence label $y^{(entity)}$ across all N words in the sentence:

$$L_{word} = -\sum_{m} \sum_{t=1}^{N} \sum_{i=1}^{k} \left[a_{ti}^{(m)} log\left(\hat{a}_{ti}^{(m)}\right)\right] \tag{21}$$

Similar to [119], we also add another loss function for joining the sentence-level and word-level objectives that encourages the model to optimize for two conditions on the ADE sentence (i) an ADE sentence must have at least one ADE entity word, and (ii) ADE sentence must have at least one word that is either non-ADE entity or a no-entity word.

$$\begin{aligned} L_{attn} = &\sum_{m} \left(\min\left(\hat{a}_{t,ADE}^{(m)}\right) - 0\right)^2 \\ &+ \\ &\sum_{m} \left(\max\left(\hat{a}_{t,ADE}^{(m)}\right) - y^{(m)}\right)^2 \end{aligned} \tag{22}$$

We combine different objective functions using weighting parameters to allow us to control the importance of each objective. The final objective that we minimize during



(a) Single Task-specific Attention

(b) Dual Task-specific attention

(c) Single Supervised Self-attention

(d) Dual Supervised Self-attention

(e) Distribution of attention weights

(f) Sentence prediction scores

Figure 15: Single v.s. dual attention distribution. The color intensity corresponds to the weight given to each word. Attention weight of each word are given in the parenthesis. Single attention-based models (a) and (c) fail to capture sufficient attention weight on the key semantic areas of the sentence. The dual-attention based model where the two attention distributions are combined, accurate weights are assigned (b) and (d).

training is then:

$$L = \lambda_{sent} \cdot L_{sent} \; + \; \lambda_{word} \cdot L_{word} \; + \; \lambda_{attn} \cdot L_{attn} \tag{23}$$

By using word-level entity predictions as attention weights for composing sentence-level representations, we explicitly connect the predictions at both levels of granularity. When both objectives work in tandem, they help improve the performance of one another. In our joint model, we give equal importance to both tasks and set $\lambda_{word} = \lambda_{sentence} = 1$.

4.4 Experimental Results

4.4.1 Data Set

MADE1.0 NLP challenge for detecting medication and ADE related information from EHR [73] used 1089 de-identified EHR notes from 21 cancer patients (Training: 876 notes, Testing: 213 notes). The annotation statistics of the corpus are provided [120].

Named Entity Labels. The notes are annotated with several categories of medication in-

formation. *Adverse Drug Event (ADE), Drugname, Indication* and *Other Sign Symptom and Diseases (OtherSSD)* are specified as medical events that contribute to a change in a patient's medical status. *Severity, Route, Frequency, Duration* and *Dosage* specified as attributes describe important properties about the medical events. *Severity* denotes the severity of a disease or symptom. *Route, Frequency, Duration* and *Dosage* as attributes of *Drugname* label the medication method, frequency of dosage, duration of dosage, and the dosage quantity, respectively.

Sentence Labels. MADE 1.0 text has each word manually annotated with ADE or medication related entity types. For words that belong to the ADE entity type, an additional relation annotation denotes if the ADE entity is an adverse side effect of the prescription of the *Drugname* entity. Since MADE 1.0 dataset does not have sentence-level annotations, we use the relation annotation with the word annotation to assign each sentence a label as ADE or nonADE. In this work, the relation labels are used only to assign the sentence labels, but they are not used in the supervised learning process.

4.4.2 Hyper-parameter Settings

The model operates on tokenized sentences. Tokens were lower-cased, while the character-level component receives input with the original capitalization to learn the morphological features of each word. As input, the pre-trained publicly available Glove word embeddings of size 300 [101]. The size of the learned character-level embedding are 100 dimensional vectors. The size of LSTM hidden layers for word-level and char-level LSTM are size 300 and 100 respectively. The hidden combined representation h_t was set to size 200; the attention weight layer e_t was set to size 100. The attention-weighted sentence representations TS_S and SS_S, are 200 dimensional vectors and therefore their combination context vector C_S is 400 dimensional. The Entity Prediction Embedding (EPE) L_S is of size k entities that are in BIO format. Hence EPE is a size 19 dimensional binary vector (eighteen entities plus the no entity tag). The final concatenated sentence-level S vector is thus size 419. To avoid over-fitting, we apply a dropout strategy [112, 113] of 0.5 for our model. All

Table 8: ADE sentence classification: F1 scores.

Model	F1
Baseline Individual Models	
LAST [123]	0.66
ATTN [122]	0.63
Baseline Joint Model	
[49]	0.61
MGADE	0.75

models were trained with a learning rate of 0.001 using Adam [114].

4.4.3 Results

4.4.4 ADE Assertive Sentence Classification

Table 8 compares our model against two baselines of individual ADE sentence classification models. (i) Similar to [123], LAST is a Bi-LSTM based sentence classification model that uses the last hidden states for sentence composition; (ii) Similar to [122], ATTN is a B-LSTM model that used simple attention weights for sentence composition. Our full model, MGADE succeeds to improve the F1 scores by 13.6% over the LAST baseline in testing. We also compare with a model similar to [49] joint-task model based on self-attention. MGADE outperforms their model by 23.0% for sentence classification.

Table 9: ADE entity recognition: F1 scores.

Model	F1
Baseline Individual Models	
Bi-LSTM [39]	0.56
Bi-LSTM + CRF [39]	0.63
Baseline Joint Model	
[49]	0.51
MGADE	0.63

Table 10: Effect of dual-attention layer. † denotes models with single-attention with Task-specific attention removed from Supervised Self-attention model, and vice versa.

	ADE Entity Recognition			**ADE Sentence Classification**		
Model	P	R	F1	P	R	F1
MGADE-SelfA †	0.58	0.52	0.55	0.84	0.55	0.67
MGADE-TaskA †	0.62	0.50	0.55	0.82	0.64	0.72
MGADE-DualA	0.68	0.55	0.61	0.87	0.65	0.74
MGADE	0.70	0.57	0.63	0.86	0.67	0.75

4.4.5 ADE Named Entity Recognition

Table 9 compares our model against the best performing models on MADE1.0 benchmark in the literature [39] for ADE entity recognition. The entity recognition component of our MGADE is similar to their Bi-LSTM model. MGADE improves the F1 score by 12.5% over their Bi-LSTM only model. Our model achieved comparable results with their Bi-LSTM + CRF combination model. The models with CRF layer predict the label sequence jointly instead of predicting each label individually which is helpful to predict sequences where the label for each word in a sequence depends on the label of the previous word. Adding an CRF component to our model might further improve the performance of the entity recognition task. We also compare with a model similar to [49] joint-task model based on self-attention. MGADE outperforms their model by 23.5% for entity recognition.

4.4.6 Ablation Analysis

To evaluate the effect of each part in our model, we remove core sub-components and quantify the performance drop in F1 score.

Types of Attention. Table 10 studies the two types of attention we generate: Supervised self-attention (β) and Task-specific attention (α) for composing sentence-level representations. † denotes the models with single-attention. As shown in the table, models that used only a single attention component, be it Supervised Self-Attention based (SS_S) or Task-specific attention based sentence representation (TS_S) achieved the same F1-score for the entity recognition task. However, their sentence classification task performance varies, demonstrating that the two attentions capture different aspects of information in the sen-

tence. The type of attention captured plays a critical role in composing an informative sentence representation. Both single-attention models performed better than the baseline individual sentence-classification models LAST and ATTN (see Table 8). TS_S achieved superior sentence classification performance over SS_S. Intuitively, stronger focus should be placed on the words indicative of the sentence type, and TS_S which emphasizes more on the parts relevant to the ADE sentence classification task is more accurate in identifying ADE sentences.

Single Attention v.s. Dual-Attention. Table 10 studies impact of dual-attention component. As seen, the model with dual-attention sentence representation which combines two attention-weighted sentence representations C_S outperforms the models with single-attention (denoted by †) in both entity recognition and sentence classification tasks.

Label-Awareness. Table 10 studies the effect of adding the label-awareness component in improving the sentence representation. Our full model MGADE, with both dual-attention and label-aware components further improves the performance of sentence classification and entity recognition tasks by 1.0% and 2.0% respectively compared to MGADE-DualA, the model with only dual-attention component.

Case Study. Dual-attention is not only effective in capturing multiple aspects of semantic information in the sentence, but also in reducing the risk of capturing incorrect or insufficient attention when only one of the single attentions (either task-specific or supervised self-attention) is used. Fig 15 shows such an example where single attention, either task-specific or supervised self-attention, fails to capture sufficient attention weight on the key semantic areas of the sentence necessary to make a correct prediction on the sentence. The incorrect distribution of attention weights assigned in the single task-specific and single supervised self-attention (Figures 15a and 15c) is addressed by the dual-attention mechanism. The later corrects the distribution and assigns appropriate weights to the relevant semantic words as in Figures 15b and 15d. In Figures 15e and 15f, we demonstrate the effectiveness of the dual-attention mechanism by plotting attention weight distributions and the sentence prediction scores when specific type of attention is composed into the

sentence representation. The bar chart depicts the ADE sentence-level classification confidence scores w.r.t single-attention and dual-attention models and confirms the utility of dual-attention.

4.5 Discussion

The primary focus for this work is on detecting ADE entity word, and ADE assertive sentences from the text. There are two reasons for focusing on the ADE entity – (i) ADEs are one of the critical information categories in the text whose detection is crucial to effective pharmacovigilance, and (ii) they are most challenging to detect due to the entity-type ambiguity problem. Moreover, the presence of ADE entity words in the sentence is a key indicator of an ADE sentence and hence required in the identification of ADE assertive sentence.

A critical problem with medical narratives is that the data for the *ADE* entity is highly skewed, that is, less than 2.5% of the words in the narratives are *ADE* entity words leading to a class imbalanced dataset. In such situations, the under-represented *ADE* entity being the minority class but still the class of primary interest is not well learned and hence often mis-classified. For that reason, our training process involves using other named entity labels linked with medication and its attributes as they provide supplementary information in the ADE sentence-level detection task. Our dual-attention component is specifically designed to capture relevant task-specific semantic areas of the sentence which, in the ADE detection task accounts for providing stronger emphasis on words that are indicative of the ADE entity type.

4.6 Summary

We propose a dual-attention network for multi-grained ADE detection to jointly identify ADE entities and ADE assertive sentences from medical narratives. Our model effectively supports knowledge sharing between the two levels of granularity, i.e., words and sentences, improving the overall quality of prediction on both tasks. Our solution

features significant performance improvements over state-of-the-art models on both tasks. Our MGADE architecture is pluggable, in that other sequential learning models including BERT [124] or other models for sequence labelling and text classification could be substituted in place of the Bi-LSTM sequential representation learning model. We leave this enhancement of our model and its study to future work.

5 Conclusions

5.1 Contributions

In this dissertation, we propose ADE information detection methodology that enables automatic identification and extraction of high value ADE related information from unstructured medical narratives. This functionality is useful in a wide range of applications in the healthcare domain. For example, our ML-NLP based information extraction methods can be utilized for (i) additional characterization of the signal from EHR narratives which aids in better understanding and analysis of the prevalence relevance of ADE signals, and (ii) vaccine safety surveillance using EHR narratives where such data may be able to highlight new risks associated with vaccine, as well as adverse events that have high background incidence rates and events that are not pharmacologically predictable. The extracted information can be further utilized by advanced data mining or visualization techniques for making the clinical review process more productive and effective. The key contributions of this dissertation can be summarized as follows.

First, we propose the Tiered Ensemble Learning System with Diversity (TELS-D) approach - a collective approach to deal with highly skewed and class imbalanced data, and the limitations of any one individual classification method . We create a balanced training environment by applying undersampling techniques. We generate an ensemble of heterogeneous classifiers in this balanced training environment leveraging an novel ensemble combiner. We train a "*learner-over-learners*" meta-algorithm over the class predictions from the heterogeneous classifiers to correctly learn and classify the named entities in the narratives. Our results are promising and indicate that, in the context of binary classification an ensemble learning approach would be a better choice for NER especially for class imbalanced datasets. Our experimental results show that : (i) a balanced learning environment along with an Ensemble of Heterogeneous Classifiers constantly improves the performance over individual base learners and, (ii) stacking-based ensemble combiner methods outperform simple Majority Voting by 0.30 F-measure.

Second, we propose Dual-Level Embedding for Adverse Drug Event Detection framework (DLADE) that is able to learn long-term dependencies from the context surrounding the words in a sequence for and NER task. The framework is developed by adapting a three-layered, deep learning RNN architecture of (1) Bi-LSTM for character-level word representation to encode the morphological features of the medical terminology, (2) Bi-LSTM for capturing the contextual information of each word within a sentence, and (3) Conditional Random Fields for the final label prediction by also considering the surrounding words. We developed a rule-based EHR text preprocessor to tackle the noisy nature of EHR narratives. Our results indicate that the integration of two widely used sequence labeling techniques that complement each other along with dual-level embedding (character-level and word-level) to represent words in the input layer results in a deep learning architecture that achieves excellent information extraction accuracy for EHR notes. Our system was ranked first for the NER task in the MADE1.0 NLP challenge with a micro-averaged F1-score of 0.8290 (Official Score).

Third, we propose the Multi-Grained ADE detection network (MGADE), a dual-attention based neural network model for jointly learning to detect ADE information from narratives at different levels of granularity (word-level and sentence-level). We designed an end-to-end joint model that leverages the inter-dependencies among these two granularities. We introduce two novel strategies of constructing multiple complementary sentence-level representations to enhance the performance of sentence classification. We utilize the dual-attention aspect of the model to provide an intuitive explanation of the model and insights on its strengths and weaknesses. Our model effectively supports knowledge sharing between the two levels of granularity, i.e., words and sentences, improving the overall quality of prediction on both tasks. Our experiments on benchmark dataset of EHR notes [120] demonstrate that our MGADE model drives up the F1-score for both tasks significantly: (i) entity recognition of ADE words by 12.5% and by 23.5% and (ii) ADE sentence classification by 13.6% and by 23.0%, compared to state-of-art single task and joint-task models, respectively.

5.2 Future Research Directions

We briefly summarize interesting research directions towards more advanced ADE information extraction.

Sequential Sentence Classification. Explore extending the sentence-level ADE detection model which classifies sentences in isolation to incorporate sequential sentence classification that utilizes the contextual information within surrounding sentences to help classify the current sentence. The structured prediction of sentences might prove to be helpful in improving the overall classification performance of ADE entity and assertive sentence classification tasks. Interesting research directions might include identifying different types of important sentences from narratives, and extending the binary-classification of sentences to a multi-class and/or multi-label sentence-level classification approach depending upon the annotated labels of interest in the dataset. We leave this enhancement of our model and its study to future work.

Focal loss. A common challenge between word-level and sentence-level ADE detection is the highly skewed training datasets leading to a class imbalance.

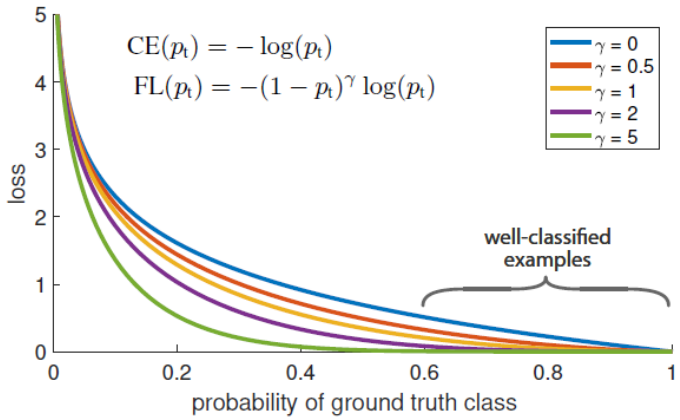


Figure 16: Focal Loss (Lin et al. 2017). CE: Standard Cross Entropy Loss. FL: Focal Loss

Explore addressing the ADE class imbalance and enhance the training process with a new loss function, "Focal Loss" (FL) as proposed by Lin et al. [125], by reshaping the

standard cross entropy loss such that it down-weights the loss assigned to well-classified easy training examples and thus focus training on hard negatives. While, adjusting class weights balances the importance of positive/negative examples, it does not differentiate between easy/hard examples. They demonstrated that FL acts as a more effective alternative to previous approaches for dealing with class imbalance. As illustrated in Figure 16, FL adds a factor $(1 - p_t)_\gamma$ to the standard cross entropy criterion. Setting $\gamma > 0$ reduces the relative loss for well-classified examples $(p_t > .5)$, putting more focus on hard, misclassified examples. The original paper demonstrated the utility of FL in the context of a binary classification problem. The complex nature of ADE information detection problem with multi-class classification approach for both word-level entity detection and sentence-level classification tasks, calls for an more elaborate study of adapting FL to ADE detection problem setting. We leave this enhancement of our model and its study to future work.

Advanced sequential models. Recently, Transformer [126] based approaches such as BERT [124], GPT-2 [127] type of language models are achieving performance improvement over the state-of-the-art RNN based architectures on several NLP tasks. Fine tuning and using these more advanced models for ADE detection, for both ADE entity recognition (sequence tagging) and ADE assertive sentence classification in place of our Bi-LSTM sequential representation learning model is worth exploring to drive performance improvements. We leave this enhancement of our model and its study to future work.

References

[1] World Health Organization et al. The importance of pharmacovigilance. 2002.

[2] Molla S. Donaldson, Janet M. Corrigan, Linda T. Kohn, and Editors. *To err is human: building a safer health system*, volume 6. National Academies Press, 2000.

[3] Institute of Medicine. Institute of medicine report from the committee on advancing pain research, care, and education: Relieving pain in america, a blueprint for transforming prevention, care, education and research, 2011.

[4] US Department of Health and Human Services. National action plan for adverse drug event prevention. *Washington, DC*, pages 50–98, 2014.

[5] Jason Lazarou, Bruce H Pomeranz, and Paul N Corey. Incidence of adverse drug reactions in hospitalized patients: a meta-analysis of prospective studies. *Jama*, 279 (15):1200–1205, 1998.

[6] AJ Weiss, WJ Freeman, KC Heslin, and ML Barrett. Adverse drug events in us hospitals, 2010 versus 2014. *HCUP Statistical Brief*, 234, 2018.

[7] Food, Drug Administration, et al. Guidance for industry: good pharmacovigilance practices and pharmacoepidemiologic assessment. *Rockville, MD: Food and Drug Administration*, 2005.

[8] Richard Platt, Marcus Wilson, K Arnold Chan, Joshua S Benner, Janet Marchibroda, and Mark McClellan. The new sentinel network—improving the evidence of medical-product safety. *New England Journal of Medicine*, 361(7):645–647, 2009.

[9] Rave Harpaz, Santiago Vilar, William DuMouchel, Hojjat Salmasian, Krystl Haerian, Nigam H Shah, Herbert S Chase, and Carol Friedman. Combing signals from spontaneous reports and electronic health records for detection of adverse drug reactions. *Journal of the American Medical Informatics Association*, 20(3):413–419, 2013.

[10] Nicholas P Tatonetti, Guy Haskin Fernald, and Russ B Altman. A novel signal detection algorithm for identifying hidden drug-drug interactions in adverse event reports. *Journal of the American Medical Informatics Association*, 19(1):79–85, 2012.

[11] FDA. FAERS (FDA adverse event reporting system), 2016. URL `\url{http://www.fda.gov/Drugs/GuidanceComplianceRegulatoryInformation/Surveillance/AdverseDrugEffects/ucm082193.htm}`.

[12] Andrew M Wilson, Lehana Thabane, and Anne Holbrook. Application of data mining techniques in pharmacovigilance. *BJCP*, 57(2):127–134, 2004.

[13] Xiaodong Feng, Amie Cai, Kevin Dong, Wendy Chaing, Max Feng, Nilesh S Bhutada, John Inciardi, and Tibebe Woldemariam. Assessing pancreatic cancer risk associated with dipeptidyl peptidase 4 inhibitors: Data mining of fda adverse event reporting system (faers). *Journal of Pharmacovigilance*, pages 1–7, 2013.

[14] Toshiyuki Sakaeda, Akiko Tamon, Kaori Kadoyama, and Yasushi Okuno. Data mining of the public version of the fda adverse event reporting system. *International journal of medical sciences*, 10(7):796, 2013.

[15] Jing Jiang. Information extraction from text. In *Mining text data*, pages 11–41. Springer, 2012.

[16] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1412–1421, 2015.

[17] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014. URL `http://arxiv.org/abs/1409.0473`.

[18] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J. Smola, and Eduard H. Hovy. Hierarchical attention networks for document classification. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 1480–1489, 2016.

[19] Chandra Pandey, Zina M. Ibrahim, Honghan Wu, Ehtesham Iqbal, and Richard J. B. Dobson. Improving RNN with attention and embedding for adverse drug reactions. In *Proceedings of the 2017 International Conference on Digital Health, London, United Kingdom, July 2-5, 2017*, pages 67–71, 2017.

[20] Fei Li, Yue Zhang, Meishan Zhang, and Donghong Ji. Joint models for extracting adverse drug events from biomedical text. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2838–2844, 2016.

[21] Taxiarchis Botsis, Michael D Nguyen, Emily Jane Woo, Marianthi Markatou, and Robert Ball. Text mining for the vaccine adverse event reporting system: medical text classification using informative feature selection. *Journal of the American Medical Informatics Association*, 18(5):631–638, 2011.

[22] Christian M. Rochefort, Aman Verma, Tewodros Eguale, Todd C. Lee, and David L. Buckeridge. A novel method of adverse event detection can accurately identify venous thromboembolisms (vtes) from narrative electronic health record data. *JAMIA*, 22(1):155–165, 2015. doi: 10.1136/amiajnl-2014-002768. URL `https://doi.org/10.1136/amiajnl-2014-002768`.

[23] Wei Wang, Krystl Haerian, Hojjat Salmasian, Rave Harpaz, Herbert Chase, and Carol Friedman. A drug-adverse event extraction algorithm to support pharmacovigilance knowledge mining from pubmed citations. In *AMIA annual symposium proceedings*, volume 2011, page 1464. American Medical Informatics Association, 2011.

[24] Shantanu Dev, Shinan Zhang, Joseph Voyles, and Anand S Rao. Automated classification of adverse events in pharmacovigilance. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 905–909. IEEE, 2017.

[25] Carol Friedman. A broad-coverage natural language processing system. In *Proceedings of the AMIA Symposium*, page 270. American Medical Informatics Association, 2000.

[26] Robert Leaman, Laura Wojtulewicz, Ryan Sullivan, Annie Skariah, Jian Yang, and Graciela Gonzalez. Towards internet-age pharmacovigilance: extracting adverse drug reactions from user posts to health-related social networks. In *Proceedings of the 2010 workshop on biomedical natural language processing*, pages 117–125. Association for Computational Linguistics, 2010.

[27] Harsha Gurulingappa, Roman Klinger, Martin Hofmann-Apitius, and Juliane Fluck. An empirical evaluation of resources for the identification of diseases and adverse effects in biomedical literature. In *2nd Workshop on Building and evaluating resources for biomedical text mining (7th edition of the Language Resources and Evaluation Conference)*, 2010.

[28] Hua Xu et al. Medex: a medication information extraction system for clinical narratives. *JAMIA*, 17(1):19–24, 2010.

[29] Adrian Benton, Lyle Ungar, Shawndra Hill, Sean Hennessy, Jun Mao, Annie Chung, Charles E Leonard, and John H Holmes. Identifying potential adverse effects using the web: A new approach to medical hypothesis generation. *Journal of biomedical informatics*, 44(6):989–996, 2011.

[30] Andrew Yates and Nazli Goharian. Adrtrace: detecting expected and unexpected adverse drug reactions from user reviews on social media sites. In *European Conference on Information Retrieval*, pages 816–819. Springer, 2013.

[31] Harsha Gurulingappa, Abdul Mateen-Rajpu, and Luca Toldo. Extraction of potential adverse drug events from medical case reports. *Journal of biomedical semantics*, 3(1): 15, 2012.

[32] Hariprasad Sampathkumar, Xue-wen Chen, and Bo Luo. Mining adverse drug reactions from online healthcare forums using hidden markov model. *BMC medical informatics and decision making*, 14(1):1, 2014.

[33] Balaji Polepalli Ramesh et al. Automatically recognizing medication and adverse event information from food and drug administration's adverse event reporting system narratives. *JMIR medical informatics*, 2(1):e10, 2014.

[34] Alec B Chapman, Kelly S Peterson, Patrick R Alba, Scott L DuVall, and Olga V Patterson. Detecting adverse drug events with rapidly trained classification models. *Drug safety*, 42(1):147–156, 2019.

[35] Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Xiangnan Kong, Elke A. Rundensteiner, Sanjay K. Sahoo, and Suranjan De. One size does not fit all: An ensemble approach towards information extraction from adverse drug event narratives. In *Proceedings of HEALTHINF*, pages 176–188. Proceedings of HEALTHINF, 2018.

[36] Abhyuday N. Jagannatha and Hong Yu. Structured prediction models for rnn based sequence labeling in clinical text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, volume 2016, page 856. Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing, 2016.

[37] Fei Li, Meishan Zhang, Guohong Fu, and Donghong Ji. A neural joint model for entity and relation extraction from biomedical text. *BMC bioinformatics*, 18(1):198, 2017.

[38] Fei Li, Weisong Liu, and Hong Yu. Extraction of information related to adverse drug events from electronic health record notes: design of an end-to-end model based on deep learning. *JMIR medical informatics*, 6(4):e12159, 2018.

[39] Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Cansu Sen, Elke A Rundensteiner, and Xiangnan Kong. Adverse drug event detection from electronic health records using hierarchical recurrent neural networks with dual-level embedding. *Drug safety*, 42(1):113–122, 2019.

[40] Steven Bird. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. ACL, 2006.

[41] Bharath Dandala, Venkata Joopudi, and Murthy Devarakonda. Adverse drug events detection in clinical notes by jointly modeling entities and relations using neural networks. *Drug safety*, 42(1):135–146, 2019.

[42] Xi Yang, Jiang Bian, Yan Gong, William R Hogan, and Yonghui Wu. Madex: A system for detecting medications, adverse drug events, and their relations from clinical notes. *Drug safety*, 42(1):123–133, 2019.

[43] Harsha Gurulingappa, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. Identification of adverse drug event assertive sentences in medical case reports. In *First international workshop on knowledge discovery and health care management (KD-HCM), European conference on machine learning and principles and practice of knowledge discovery in databases (ECML PKDD)*, pages 16–27, 2011.

[44] Azadeh Nikfarjam, Abeed Sarker, Karen O'Connor, Rachel Ginn, and Graciela Gonzalez. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, 22(3):671–681, 2015.

[45] Abeed Sarker and Graciela Gonzalez. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *Journal of biomedical informatics*, 53:196–207, 2015.

[46] Trung Huynh, Yulan He, Alistair Willis, and Stefan Rüger. Adverse drug reaction classification with deep neural networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 877–887, 2016.

[47] Ahmad P Tafti, Jonathan Badger, Eric LaRose, Ehsan Shirzadi, Andrea Mahnke, John Mayer, Zhan Ye, David Page, and Peggy Peissig. Adverse drug event discovery using biomedical literature: a big data neural network adventure. *JMIR medical informatics*, 5(4):e51, 2017.

[48] Rich Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, Jul 1997. ISSN 1573-0565. doi: 10.1023/A:1007379606734. URL https://doi.org/10.1023/A:1007379606734.

[49] Shinan Zhang, Shantanu Dev, Joseph Voyles, and Anand S Rao. Attention-based multi-task learning in pharmacovigilance. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 2324–22328. IEEE, 2018.

[50] Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Xiangnan Kong, and Elke A. Rundensteiner. A dual-attention network for joint named entity recognition and sentence classification of adverse drug events. *Findings of EMNLP, Empirical Methods in Natural Language Processing*, 2020.

[51] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *Advances in neural information processing systems*, pages 2204–2212, 2014.

[52] Feifan Liu, Abhyuday Jagannatha, and Hong Yu. Towards drug safety surveillance and pharmacovigilance: current progress in detecting medication and adverse drug events from electronic health records, 2019.

[53] Mikel Galar, Alberto Fernandez, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484, 2012.

[54] Susmitha Wunnava, Xiao Qin, Tabassum Kakar, ML Tlachac, Xiangnan Kong, Elke A Rundensteiner, Sanjay K Sahoo, and Suranjan De. Multi-layered learning for information extraction from adverse drug event narratives. In *International Joint Conference on Biomedical Engineering Systems and Technologies*, pages 420–446. Springer, 2018.

[55] Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Vimig Socrates, Amber Wallace, and Elke Rundensteiner. Towards transforming fda adverse event narratives into actionable structured data for improved pharmacovigilance. In *Proceedings of the Symposium on Applied Computing*, pages 777–782. ACM, 2017.

[56] Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Elke A. Rundensteiner, and Xiangnan Kong. Deep learning strategies for the automatic detection of medication and adverse drug events from electronic health records. In *Proceedings of the American Medical Informatics Association Annual Symposium (AMIA)*, 2018.

[57] Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Elke A. Rundensteiner, and Xiangnan Kong. Bidirectional lstm-crf for adverse drug event tagging in electronic health records. volume 90, page 48–56. Proceedings of the 1st International Workshop on Medication and Adverse Drug Event Detection, volume 90 of Proceedings of Machine Learning Research, May 2018. URL `http://proceedings.mlr.press/v90/wunnava18a.html`.

[58] Zachary Chase Lipton. A critical review of recurrent neural networks for sequence learning. *CoRR*, abs/1506.00019, 2015. URL `http://arxiv.org/abs/1506.00019`.

[59] Sepp Hochreiter and Jurgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–1780, 1997.

[60] Council for International Organizations of Medical Sciences, Council for International Organizations of Medical Sciences, et al. Practical aspects of signal detection in pharmacovigilance: report of cioms working group viii. *Geneva: Council for International Organizations of Medical Sciences*, 2010.

[61] Preciosa M Coloma, Gianluca Trifirò, Vaishali Patadia, and Miriam Sturkenboom. Postmarketing safety surveillance: where does signal detection using electronic healthcare records fit into the big picture? *Drug safety*, 36(3):183–197, 2013.

[62] Vassilis G Koutkias and Marie-Christine Jaulent. Computational approaches for pharmacovigilance signal detection: toward integrated and semantically-enriched frameworks. *Drug safety*, 38(3):219–232, 2015.

[63] Tracy A Lieu, Martin Kulldorff, Robert L Davis, Edwin M Lewis, Eric Weintraub, Katherine Yih, Ruihua Yin, Jeffrey S Brown, Richard Platt, and Vaccine Safety Datalink Rapid Cycle Analysis Team. Real-time vaccine safety surveillance for the early detection of adverse events. *Medical care*, pages S89–S95, 2007.

[64] Moncef Slaoui and Matthew Hepburn. Developing safe and effective covid vaccines—operation warp speed's strategy and approach. *New England Journal of Medicine*, 2020.

[65] Alan R Aronson. Effective mapping of biomedical text to the umls metathesaurus: the metamap program. In *Proceedings of the AMIA Symposium*, page 17. AMIA, 2001.

[66] Özlem Uzuner, Imre Solti, and Eithon Cadag. Extracting medication information from clinical text. *JAMIA*, 17(5):514–518, 2010.

[67] Guergana K Savova et al. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. *JAMIA*, 17(5): 507–513, 2010.

[68] David Ferrucci and Adam Lally. Uima: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348, 2004.

[69] Carol Friedman, Philip O Alderson, John HM Austin, James J Cimino, and Stephen B Johnson. A general natural-language text processor for clinical radiology. *JAMIA*, 1 (2):161–174, 1994.

[70] Özlem Uzuner, Xiaoran Zhang, and Tawanda Sibanda. Machine learning and rule-based approaches to assertion classification. *JAMIA*, 16(1):109–115, 2009.

[71] Omid Ghiasvand. *Disease name extraction from clinical text using conditional random fields*. PhD thesis, The University of Wisconsin-Milwaukee, 2014.

[72] Scott Russell Halgrim, Fei Xia, Imre Solti, Eithon Cadag, and Özlem Uzuner. A cascade of classifiers for extracting medication information from discharge summaries. *Journal of biomedical semantics*, 2(3):S2, 2011.

[73] Abhyuday N Jagannatha and Hong Yu. Bidirectional rnn for medical event detection in electronic health records. In *Proceedings of the conference. ACL. North American Chapter. Meeting*, volume 2016, page 473. NIH Public Access, 2016.

[74] Son Doan and Hua Xu. Recognizing medication related entities in hospital discharge summaries using support vector machine. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 259–266. ACL, 2010.

[75] Rushi Longadge and Snehalata Dongre. Class imbalance problem in data mining review. *arXiv preprint arXiv:1305.1707*, 2013.

[76] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intelligent data analysis*, 6(5):429–449, 2002.

[77] Christopher M Bishop. Pattern recognition. *Machine Learning*, 128, 2006.

[78] Rave Harpaz, Alison Callahan, Suzanne Tamang, Yen Low, David Odgers, Sam Finlayson, Kenneth Jung, Paea LePendu, and Nigam H Shah. Text mining for adverse drug events: the promise, challenges, and state of the art. *Drug safety*, 37(10):777–790, 2014.

[79] Özlem Uzuner, Imre Solti, Fei Xia, and Eithon Cadag. Community annotation experiment for ground truth generation for the i2b2 medication challenge. *JAMIA*, 17 (5):519–523, 2010.

[80] Steven Bird et al. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[81] Eugene Charniak and Mark Johnson. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 173–180. ACL, 2005.

[82] Michael Kuhn, Ivica Letunic, Lars Juhl Jensen, and Peer Bork. The sider database of drugs and side effects. *Nucleic acids research*, 44(D1):D1075–D1079, 2015.

[83] Ethem Alpaydin. *Introduction to machine learning*. MIT press, 2014.

[84] R. Polikar. Ensemble learning. *Scholarpedia*, 4(1):2776, 2009. doi: 10.4249/scholarpedia.2776. revision #91224.

[85] Pang-Ning Tan et al. *Introduction to data mining*. Pearson Education India, 2006.

[86] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[87] Robert E Schapire. The strength of weak learnability. *Machine learning*, 5(2):197–227, 1990.

[88] David H Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

[89] Hoang Nguyen and Jon Patrick. Text mining in clinical domain: Dealing with noise. In *KDD*, pages 549–558, 2016.

[90] Nitesh V Chawla. Data mining for imbalanced datasets: An overview. In *Data mining and knowledge discovery handbook*, pages 875–886. Springer, 2009.

[91] Jerzy Błaszczyński, Jerzy Stefanowski, and Łukasz Idkowiak. Extending bagging for imbalanced data. In *Proceedings of the 8th International Conference on Computer Recognition Systems CORES 2013*, pages 269–278. Springer, 2013.

[92] Ricardo Barandela, Rosa Maria Valdovinos, and José Salvador Sánchez. New applications of ensembles of classifiers. *Pattern Analysis & Applications*, 6(3):245–256, 2003.

[93] Shuo Wang and Xin Yao. Diversity analysis on imbalanced data sets by using ensemble models. In *Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining, CIDM*, pages 324–331, 2009.

[94] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Stanford, CA, 1995.

[95] Louise Deleger, Cyril Grouin, and Pierre Zweigenbaum. Extracting medical information from narrative patient records: the case of medication-related information. *Journal of the American Medical Informatics Association*, 17:555–558, 2010.

[96] John Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[97] Elena Tutubalina and Sergey Nikolenko. Combination of deep recurrent neural networks and conditional random fields for extracting adverse drug reactions from user reviews. *Journal of Healthcare Engineering*, 2017, 2017.

[98] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[99] Sebastien Dubois and Nathanael Romano. Learning effective embeddings from medical notes.

[100] Youngduck Choi, Chill Yi-I. Chiu, and David Sontag. Learning low-dimensional representations of medical concepts. *AMIA Summits on Translational Science Proceedings*, page 41, 2016.

[101] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543. Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014.

[102] Donald C. Comeau, Rezarta Islamaj Dogan, Paolo Ciccarese, Kevin Bretonnel Cohen, Martin Krallinger, Florian Leitner, Zhiyong Lu, Yifan Peng, Fabio Rinaldi, Manabu Torii, et al. Bioc: a minimalist approach to interoperability for biomedical text processing. *Database*, 2013, 2013.

[103] Steven Bird and Edward Loper. Nltk: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 31. Proceedings of the ACL 2004 on Interactive poster and demonstration sessions, 2004.

[104] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60. Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations, 2014.

[105] Lance A. Ramshaw and Mitchell P. Marcus. *Text chunking using transformation-based learning*. Springer, 1999.

[106] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537, 2011.

[107] Cicero D. Santos and Bianca Zadrozny. Learning character-level representations for part-of-speech tagging. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1818–1826. Proceedings of the 31st International Conference on Machine Learning (ICML-14), 2014.

[108] Patrice Bengio and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5:157–166, 1994.

[109] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning*, pages 1310–1318. International Conference on Machine Learning, 2013.

[110] Felix A. Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. *IET*, 1999.

[111] Mike Schuster and Kuldip K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45:2673–2681, 1997.

[112] Xuezhe Ma and Eduard H. Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers, 2016.

[113] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15:1929–1958, 2014.

[114] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[115] Martın Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283. OSDI, 2016.

[116] Sampo Pyysalo, F. Ginter, Hans Moen, T. Salakoski, and Sophia Ananiadou. Distributional semantics resources for biomedical text processing. January 2013.

[117] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[118] Lemao Liu, Masao Utiyama, Andrew M. Finch, and Eiichiro Sumita. Neural machine translation with supervised attention. In Nicoletta Calzolari, Yuji Matsumoto, and Rashmi Prasad, editors, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 3093–3102. ACL, 2016. URL `https://www.aclweb.org/anthology/C16-1291/`.

[119] Marek Rei and Anders Søgaard. Jointly learning to label sentences and tokens. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6916–6923, 2019.

[120] Abhyuday Jagannatha, Feifan Liu, Weisong Liu, and Hong Yu. Overview of the first natural language processing challenge for extracting medication, indication, and adverse drug events from electronic health record notes (made 1.0). *Drug safety*, 42(1): 99–111, 2019.

[121] Peng Wang, Lingqiao Liu, Chunhua Shen, Zi Huang, Anton van den Hengel, and Heng Tao Shen. Multi-attention network for one shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2721–2729, 2017.

[122] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489, 2016.

[123] Franck Dernoncourt, Ji Young Lee, and Peter Szolovits. Neural networks for joint sentence classification in medical paper abstracts. In Mirella Lapata, Phil Blunsom, and Alexander Koller, editors, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, pages 694–700. Association for Computational Linguistics, 2017. doi: 10.18653/v1/e17-2110. URL https://doi.org/10.18653/v1/e17-2110.

[124] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423. URL https://doi.org/10.18653/v1/n19-1423.

[125] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(2):318–327, 2020. doi: 10.1109/TPAMI.2018.2858826. URL https://doi.org/10.1109/TPAMI.2018.2858826.

[126] 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*, pages 5998–6008, 2017.

[127] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1 (8):9, 2019.