Extracting and Grouping Medication Information with Medical Information Boundary Detection and Domain Specific BERT Named Entity Recognition

Collin Cunningham
School of Information
University of California - Berkeley
collincun@berkeley.edu

Siddartha Jakkamreddy
School of Information
University of California - Berkeley
jakkamreddy@berkeley.edu

Abstract

The most voluminous body of medical data is free-form doctor's notes. Extracting meaningful data through heuristics and rule-based approaches is a nearly impossible task given the diversity of formats, cataphora, and spellings (including errors). Neural named entity extraction (hereon NER) has demonstrated measurable success in extracting information. We seek to outline a two-pronged approach to fulfill the goal of NER and relation extraction (RE): firstly by chunking relevant information, and secondly using domain specific BERT models specifically trained on medical data for state-of-the-art NER performance.

1 Introduction

Electronic health records (EHR) are used to store the data generated from doctor-patient interactions. Even while many modern EHRs have nurses input structured medication and diagnoses, doctors still commonly write free-form text about their interaction with the patient which may or may not be included in the nurse patient interaction (Murdoch and Detsky, 2013). For example, the nurse may ask about prior diagnoses storing this information in a database while the doctor will diagnosis the patient with a new disease which is left in the doctor's note. Upon the next interaction, the doctor must read both to make sure nothing is missed.

The companies who engineer EHRs have attempted extracting entities from text with rules-based methodologies. As an example, they compare text with hard-coded lists of medications—not only is this not performant but it must be constantly maintained and disallows any error or new abbreviations and shorthands. Since the early part of this decade (Meystre et al., 2007), researchers have begun exploring natural language processing (NLP) techniques for NER including deep learning (Wu et al., 2015) (Chalapathy, Zare Borzeshi, and Piccardi, 2016) (Boag et al., 2018). However, based on the literature review of clinical information extraction (Wang et al., 2017) , most of the applications used in the health domain is heuristic focused. We attempt to provide a solution using Neural Natural Language processing models that will both extract the medications with state-of-the-art performance and attempt to connect relevant entities. We will first look at the data being used for our study in the next section.

2 Background

2.1 Dataset

The data for this project was gathered by the NIH-funded i2b2: Informatics for Integrating Biology and the Bedside and was used in the 2009 Medication Challenge. In order to access these notes, one needs to register at the i2b2 website (https://www.i2b2.org/NLP/DataSets/) and submit a proposal which is then reviewed by the i2b2 organizers. It has around 261 discharge summaries which have been annotated with medication, dosage, method of medication, frequency, duration, and reason by teams across the globe during the course of the challenge. The total number of annotations are listed below:

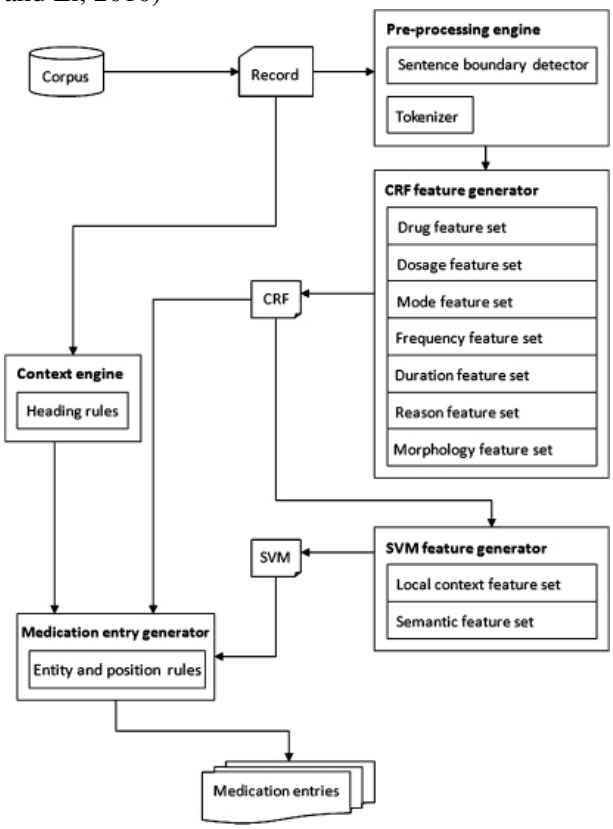
- Medication annotations: 9318
- Dosage annotations: 4666
- Mode annotations: 3513
- Frequency annotations: 4229
- Duration annotations: 571
- Reason annotations: 1694

The data itself consists of the text of the summary and an annotation file containing the list of entities, their row position and the corresponding line offset.

2.2 Medical Named Entity Recognition

We begin with the more self-explanatory of the two components of our model: medication named entity recognition. Figure 1 shows the architecture that was state-of-the-art prior to the application of deep learning (Patrick and Li, 2010).

Figure 1: Architecture of baseline model (Patrick and Li, 2010)



As shown in the architecture, This model was based on a cascaded approach, which integrated conditional random fields, support vector machines and several rule-based engines.

After 2010, most publications on the subject involve deep learning such as (Wu et al., 2015; Chalapathy, Zare Borzeshi, and Piccardi, 2016; Boag et al., 2018). These networks use GloVe (Pennington, Socher, and Manning, 2014) embeddings with Bidirectional LSTMs and the scores range between 83 and 84%. There has been further improvements using ensembles of ELMO(Peters et al., 2018) embedding with LSTMs models (Si et al., 2019). The introduction of BERT(Devlin et al., 2018) there has been a plethora of experimentation leveraging transfer learning.

2.3 Grouping Information

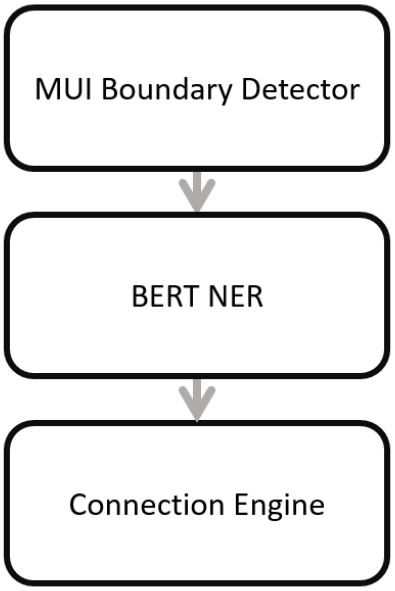
One aspect that most of the above papers fail to address is connecting related information. The goal here is to take named entities for dosage, e.g. 100 mg, and connect that with the medication entity, e.g. Aspirin, resulting in 100 mg of Aspirin. There are multiple papers regarding analogy and relation extraction which would achieve this goal (GuoDong et al., 2005). We will not be using this approach as our data was not annotated for entity relationships. However, there have been several papers on grouping information into sentences (Gillick, 2009). In fact, the solution that won the 2009 challenge (Patrick and Li, 2010) uses sentence boundary extraction to prepare for ingestion into their model. We will leverage these techniques in our model.

3 Model

3.1 Overview

Our model has three primary components as shown in Figure 2: Medical unit of information boundary detector, named entity recognition with BERT, and finally the connection engine. We will outline these components in depth below.

Figure 2: Model architecture.



3.2 MUI Boundary Detector

In order to group information, we define a medial unit of information (hereon MUI) as a single block of information that will contain all relevant medication name, dosage, etc. We achieve by training a model on a corpus of sentences that is geared towards doctor's notes. The following sections provides details on our implementation.

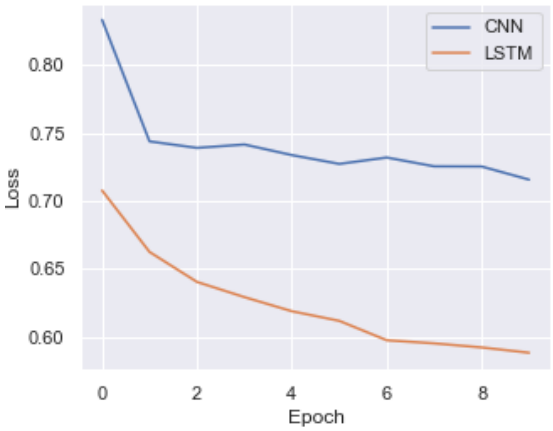
3.2.1 Baseline

The baseline model for this is a sentence boundary detector. This was a LSTM network on top of a character to number embedding that was trained on the Europarl dataset (Koehn, 2004) which resulted in an F1 score of 40%. The code for this was adapted from a GitHub repository called `deep-eos` that provided an extraction of all punctuation with a window of width w centered at the punctuation mark which severed as context.

3.2.2 Implementation

To account for the formatting and hard-coded word wrap found in doctor's notes, we took our primary dataset and added a dummy character ($\sim$) for these formatting mechanisms. At the base of our architecture, we used the same embedding as above—mapping all punctuation this time including our new dummy variable. We followed this with with our primary layer in which we tested both an LSTM and a CNN. Finally, there is a dense sigmoid layer with a binary output: whether or not this was the end of an MUI.

Figure 3: Comparison of loss over training epochs.



3.2.3 Results

We pretrained the model with the baseline model's dataset (ibid.) then used 5000 examples we annotated by hand from our I2B2 dataset to fine-tune our model to MUIs. We began by attempting to freeze all layers but the final sigmoid layer, but this model did not have the capacity to match the problem. Finally we unfroze all layers and found that the LSTM greatly outperformed the CNN (Figure 2). After some manual hyperparameter testing, we resulting in an LSTM model with an F1 score of 58.9% which is a significant improvement over our baseline model—it may seem low, but there are many unique examples of formatting that do not contain medication, and, with more time, we would limit our data to rows containing medication.

We tried numerous other approaches to solve this problem such as as augmented neural co-relation matrix (this did not have capacity to solve our problem), analogy extraction, and relation extraction which took up most of the semester. The former did not have the capacity and was unsuited for the problem, and we could not annotate enough data in time for the latter two.

3.3 Named Entity Recognition

3.3.1 Baseline

We use the first place model in the 2009 I2B2 challenge (Patrick and Li, 2010) as a baseline, as it was state-of-the-art prior to the use of deep learning. The model outlined in this paper achieved an F1 score of 85.6%.

3.3.2 The BERTs

For the sake of brevity, we will skip an in-depth explanation of BERT (Devlin et al., 2018). However, we must state that its representations of language are much deeper than that of GloVe, and it is more capable of understanding context than even bidirectional LSTMs (Devlin 2018) (We tested this and our trained BiLSTM-CNN had an F1 score of 85.6%.) BERT has been pretrained over billions of words in context which is then fine-tuned towards a specific task. BERT is a great candidate for medical named entity recognition for these reasons. However, medical language is in many ways different from normal language used on Wikipedia. Therefore, we tested a series of BERT models trained on language relating to medical, clinical studies, science, and more. A list of the BERT implementations we tried and their description are below:

- Base BERT Cased(ibid.): the standard BERT

implementation described above with the additional corpus of BookCorpus

- BioBERT(Lee et al., 2019): initialized with BERT weights then again pretrained on biomedical texts such as PubMed and PMC

- SciBERT(Beltagy, Cohan, and Lo, 2019): initialized with BERT weights then trained on academic papers from semanticscholar.org

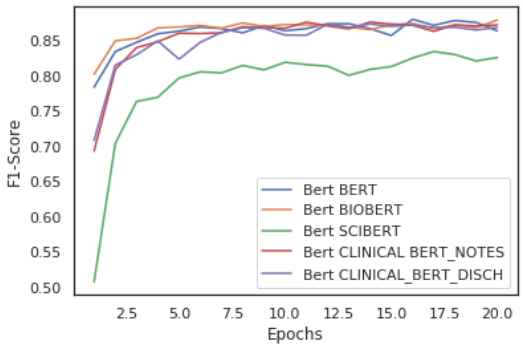
- ClinicalBERT(Alsentzer et al., 2019) (notes and discharge): both of these instances are initialized with BERT weights then pretrained using the millions of records in the MIMIC-III v1.4 database; the former is limited to discharge summaries and the latter is exclusively doctor's notes

3.3.3 Implementation

We built a custom parser to pre-process the data to add parts of speech tags and the corresponding NER tags for each word.

None of the domain specific BERT models listed above were available in Tensorflow HUB. As a result we had to use something other than Keras for our implementation. Due to the pythonic nature of PyTorch, we used PyTorch for our various BERT fine-tuning processes. We first build a process to convert tensorflow checkpoints, their corresponding vocabulary to pytorch models. We used Huggingface packages along with our custom generated PyTorch weights to train and evaluate models. This analysis was done on Tesla P100-PCIE-16GB GPU, the code for all these processes is available in the GitHub repository.

Figure 4: Comparison of F1 scores over training epochs.



3.3.4 Results

We evaluate the NLP components of our model with F1 score, as it forces a balance of precision and recall, both of which are critical at each step. BioBERT, and both ClinicalBERTs performed similarly at the top of the group with F1 scores all just about at 88%. BERT also performed well, but was consistently slightly lower as expected. Surprisingly, Scientific BERT performed worst of all with an F1 score below even our baseline model (see Figure 4). This could be attributed to the fact that Scientific BERT is pretrained on Medical Journals which adhere to strict syntactic and linguistic patterns which is not typically the case with doctor notes or summaries.

Model	F1-Score
Baseline	85.65
BiLSTM-CNN	85.6
BERT Cased - Base	87.1
BioBERT Cased	**88.0**
ScientificBERT	83.6
ClinicalBERT (Discharge)	87.9
ClinicalBERT (Notes)	**88.2**

3.4 Connection Engine

The last component of our architecture is a connection engine, which handles multiple medical terms in a single MUI. This is a simple heuristic that should be replaced in future implementations. This Python class fills columns of a table with the different named entities. Each row corresponds to a set of related information about a medication (i.e. name, dosage, duration, etc.). It fills the rows sequentially, starting a new row any time a new instance of an already filled named entity is encountered.

3.5 Inference Engine

We package all the pieces of the architecture to build out an inference solution which takes in the medical notes as input and provides a tabular output with each row consisting of related medical entities.

This has three primary components, the first component uses our saved sentence boundary detection model to identify MUIs. The second component uses our saved best model from our fine

tuning task to provide a set of word pieces and their corresponding labels. We built a process to combine word pieces to reconstruct the word to generate a list of words and their corresponding named entity labels. And finally the connection engine which uses the output from our inference to piece together different named entities.

3.6 Complete Model Results

Because we do not have annotations that tie together medication information, we cannot empirically test this complete architecture. In anecdotal tests, it performs well. Below we show an example of the entire architecture working with the input text: "The patient was given Plavix (clopidogrel) 75 mg po daily. Advised ATENOLAL 50 mg po weekly for the patient."

	B-do	B-f	B-m	B-mo
0	75mg	daily.	Plavix(clopidogrel)	po
1	50mg	weekly	ATENOLAL	po

4 Conclusion

Through this project, we have developed a novel solution for parsing unstructured doctor's notes into structured medication information. Previous models outlined in related literature solely provided semi-structured information, while we provide a fully structured tabular output.

We have been able to test the importance of context specific pretraining for BERT models where the domain specific models outperform the traditional BERT (Devlin et al., 2018) model. We would further like to evaluate other flavors of BERT like RoBERTa (Liu et al., 2019) and ALBERT (Lan et al., 2019) once their NER classification task modules are available.

While our model performs well at each step, there are opportunities for further improvement. By annotating our data (significant manual effort) to connect all relevant information in a tabular format, we would be able to completely replace the MUI boundary detector and connection engine with a single relation extractor. Moreover, this could run in parallel to the BERT NER which could allow for performance speedups as we will be avoiding the sequential step of creating a queue of MUIs as output to the BERT NER inference processes. This would also simplify our overall architecture, as loading data into the connection is fairly manual at this point.

With the results obtained within our experiments, we believe we have demonstrated very optimistic results with a clear path for further exploration in this field.

References

Koehn, Philipp (Nov. 2004). "EuroParl: A parallel corpus for statistical machine translation". In: 5.

GuoDong, Zhou et al. (2005). "Exploring Various Knowledge in Relation Extraction". In: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. ACL '05. Ann Arbor, Michigan: Association for Computational Linguistics, pp. 427–434. DOI: 10.3115/1219840.1219893.

Meystre, Stephane et al. (Nov. 2007). "Extracting Information From Textual Documents in the Electronic Health Record: A Review of Recent Research". In: *Yearb Med Inform*, pp. 128–144. DOI: 10.1055/s-0038-1638592.

Gillick, Dan (June 2009). "Sentence Boundary Detection and the Problem with the U.S." In: *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*. Boulder, Colorado: Association for Computational Linguistics, pp. 241–244.

Patrick, Jon and Min Li (Sept. 2010). "High accuracy information extraction of medication information from clinical notes: 2009 i2b2 medication extraction challenge". In: *Journal of the American Medical Informatics Association* 17.5, pp. 524–527. ISSN: 1067-5027. DOI: 10.1136/jamia.2010.003939. eprint: http://oup.prod.sis.lan/jamia/article-pdf/17/5/524/5940629/17-5-524.pdf.

Murdoch, Travis B. and Allan S. Detsky (Apr. 2013). "The Inevitable Application of Big Data to Health Care". In: *JAMA* 309.13, pp. 1351–1352. ISSN: 0098-7484. DOI: 10.1001/jama.2013.393. eprint: https://jamanetwork.com/journals/jama/articlepdf/1674245/jvp130007_1351_1352.pdf.

Pennington, Jeffrey, Richard Socher, and Christopher Manning (Oct. 2014). "Glove: Global Vec-

tors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1532–1543. DOI: 10.3115/v1/D14-1162.

Wu, Yonghui et al. (Aug. 2015). "Named Entity Recognition in Chinese Clinical Text Using Deep Neural Network". In: *Studies in health technology and informatics* 216, pp. 624–8.

Chalapathy, Raghavendra, Ehsan Zare Borzeshi, and Massimo Piccardi (Dec. 2016). "Bidirectional LSTM-CRF for Clinical Concept Extraction". In: *Proceedings of the Clinical Natural Language Processing Workshop (ClinicalNLP)*. Osaka, Japan: The COLING 2016 Organizing Committee, pp. 7–12.

Wang, Yanshan et al. (Nov. 2017). "Clinical Information Extraction Applications: A Literature Review". In: *Journal of Biomedical Informatics* 77, pp. -. DOI: 10.1016/j.jbi.2017.11.011.

Boag, Willie et al. (2018). "CliNER 2.0: Accessible and Accurate Clinical Concept Extraction". In: *CoRR* abs/1803.02245. arXiv: 1803.02245.

Devlin, Jacob et al. (2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *CoRR* abs/1810.04805. arXiv: 1810.04805.

Peters, Matthew E. et al. (2018). "Deep contextualized word representations". In: *CoRR* abs/1802.05365. arXiv: 1802.05365.

Alsentzer, Emily et al. (2019). "Publicly Available Clinical BERT Embeddings". In: *CoRR* abs/1904.03323. arXiv: 1904.03323.

Beltagy, Iz, Arman Cohan, and Kyle Lo (2019). "SciBERT: Pretrained Contextualized Embeddings for Scientific Text". In: *CoRR* abs/1903.10676. arXiv: 1903.10676.

Lan, Zhenzhong et al. (Sept. 2019). "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations". In: *arXiv e-prints*, arXiv:1909.11942, arXiv:1909.11942. arXiv: 1909.11942 [cs.CL].

Lee, Jinhyuk et al. (Jan. 2019). "BioBERT: pre-trained biomedical language representation model for biomedical text mining". In:

Liu, Yinhan et al. (July 2019). "RoBERTa: A Robustly Optimized BERT Pretraining Approach". In: *arXiv e-prints*, arXiv:1907.11692, arXiv:1907.11692. arXiv: 1907.11692 [cs.CL].

Si, Yuqi et al. (2019). "Enhancing Clinical Concept Extraction with Contextual Embedding". In: *CoRR* abs/1902.08691. arXiv: 1902.08691.