IBM Research System at TAC 2018: Deep Learning architectures for Drug-Drug Interaction extraction from Structured Product Labels

Bharath Dandala
IBM Research
Yorktown Heights, NY.
bdand@us.ibm.com

Diwakar Mahajan
IBM Research
Yorktown Heights, NY
dmahaja@us.ibm.com

Ananya Poddar
IBM Research
Yorktown Heights, NY.
poddar@us.ibm.com

Abstract

Identifying Drug-Drug Interaction (DDI) is a critical issue in clinical medicine. There is an overwhelming amount of information available in the form of narrative text in Structured Product Labeling (SPL) documents. Transforming the narrative text to structured information encoded in national standard terminologies is a prerequisite to the effective deployment of drug safety information. However, manual curation is a necessary but expensive step in the development of electronic drug-drug interaction data. To address this challenge, *TAC 2018 Drug-Drug Interaction Extraction track* defined several Natural language processing (NLP) tasks ultimately leading to identification of distinct interactions in a SPL document and linking them to structured knowledge sources. We participated in all NLP tasks namely, Concept Extraction of Precipitants, Specific-Interactions and Triggers (Task 1), Relation Identification between interactions (Task 2), normalization of different concepts (Task 3) and finally generating a global list of distinct interactions for each SPL (Task 4). We used a combination of BiLSTM-CRF followed by syntactic tree pattern matching for Task 1, Attention-LSTM for Task 2 and learning to rank algorithms for Task 3 & 4. Our system achieved F-measures of 29.87 and 35.38 for Task 1, 19.66 and 19.43 for Task 2, 24.73 and 28.37 for Task 3, 12.42 and 10.49 for Task 4 on Test 1 and Test 2 datasets respectively. We were the only team that participated in all the four tasks and we ranked among top-3 systems in Task 1 and Task 2.

Index Terms: Deep Learning, Adverse Drug Reaction, Drug Labels, Named Entity Recognition, Relation Extraction, BiLSTM-CRF, Attention-BiLSTM

1 Introduction

Structured product labels (SPLs) are a Health level Seven (HL7) standard which refer to prescription drug documents containing discrete, coded, computer-readable data, made available to the public in individual SPL index documents. They're strictly regulated by the United States Food and Drug Administration and provide critical information which health-care investors use to evaluate company's products. In SPLs, DDI details are given in narrative text, tables, and figures within the *Drug Interactions* section or other locations throughout the label. In a continued effort to transform the narrative text to structured information encoded in national standard terminologies, followed by TAC-2017, FDA and NLM organized TAC-2018 Drug-Drug Interaction challenge which includes the 4 tasks listed below:

- Task 1: Extract Mentions of *Precipitant* (Interacting drugs or substances), *Trigger* (a trigger phrase for an interaction event) and *Specific Interaction* (results of an interaction event) at a sentence level.
- Task 2: Identify interactions at a sentence level, including the Precipitant, the Specific Interaction types, namely *Pharmacokinetic, Pharmacodynamic* or *Unspecified*, and the outcomes of Pharmacokinetic and Pharmacodynamic interactions.
- Task 3: The interacting substance should be normalized to a UNII code, and the drug classes to NDF-RT NUI. Normalize the outcome of the interaction to a SNOMED CT code if it is a medical condition. Normalize pharmacokinetic effects to National Cancer Institute (NCI) Thesaurus codes.

- Task 4: Generate a global list of distinct interactions for the label in normalized form. Specifically, the interaction type, precipitant code and outcome.

Three important research areas in the field of information extraction are Named Entity Recognition (NER), Relation Extraction and Entity Linking. Extracting and succinctly linking clinical or biomedical entities automatically from the corpora has the potential to improve clinical decision support systems. With the recent advancements in deep learning research, several neural network architectures have been successfully applied to NER and Relation Extraction. In this research, we used a combination of BiLSTM-CRF followed by syntactic tree pattern matching for NER (Task 1), Attention-LSTM for Relation Extraction (Task 2) and learning to rank algorithms with manually engineered features for entity linking (Task 3 and Task 4).

The rest of the paper is organized as follows: in Section 2, we describe the dataset. In Section 3, we present our system architecture and methods for the concept and relation extraction tasks. In Section 4, we describe experimental settings of the system and achieved results for different settings and parameters. In Section 5, we conclude with our insights and details about the future direction.

2 Datasets and Preprocessing

In a collaborative effort, NLM and FDA manually annotated 22 drug label documents (XML-22) with mentions (Precipitants, Trigger, Specific Interactions), relations (Pharmacokinetic, Pharmacodynamic, Unspecified) and Interaction mappings using semi-automated approaches. Additionally, 180 drug-label documents were fully manually annotated by NLM (NLM-180) in a comparable format. We provide detailed description of the datasets in Table 1. In an effort to completely leverage the NLM-180 dataset, given it's significantly larger size, we tried to map the NLM-180 dataset to XML-22 dataset. Although mapping the datasets at a label level looks trivial, the underlying guidelines vary significantly. Thus, in order to achieve uniformity between the datasets and consequently develop our systems on these datasets, we further performed several preprocessing steps as shown below:

- It is worth noting that NLM-180 guidelines explicitly specify the annotation of only certain BiomedicalEntity (*Drug*, *Drug Class* or *Substance*) that follow specific rules. To achieve uniformity, we first automatically expanded the BiomedicalEntity annotations to cover all the drugs, drug classes or substances present in the NLM-180 dataset using classic Bidirectional Maximum Matching (BDMM) algorithm.

- Next, it is worth noting, Triggers that co-occur with Specific interactions are a separate annotation in XML-22. However such a separation does not occur in the NLM-180 dataset. Thus, to further achieve uniformity between the datasets, we converted all the discontiguous spans to contiguous spans thus leaving the bifurcation of Triggers from Specific Interactions as a post-processing step.

- Next, we identified a list of triggers, that were annotated as a Trigger in the XML-22 dataset, but not in the Specific Interaction spans in the NLM-180 dataset. We used BDMM algorithm to further expand these spans.

In Figure 1 we provide an illustration of the above method. Post transformation we were able to map NLM-180 to XML-22 as shown in Table 2.

3 Deep Learning Architectures

3.1 Concept Extraction Architecture

With the recent advancements in deep learning research, several neural network architectures have been successfully applied to concept and relation extraction. Among these, architectures based on bidirectional LSTMs have proven to be very effective (Huang et al., 2015; Ma and Hovy, 2016; Zhou et al., 2016; Zhang and Wang, 2015). In this section, we describe our concept and relation extraction systems in detail. The architectures of our concept and relation extraction systems are illustrated in Figure 2 and Figure 3 respectively.

Long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) is a type of recurrent neural network (RNN) that models interdependencies in sequential data and addresses the vanishing or exploding gradients (Bengio et al., 1994) problem of vanilla RNNs by using adaptive gating mechanism.

Given an input sequence x=$(x_1, x_2 ... x_T)$ where T is the sequence length, LSTM hidden state at

XML-22 Mentions		
Precipitant	2945	Interacting substance with a label drug i.e. drug, drug class or non-drug substance.
Trigger	793	Trigger word or phrase for an interaction event.
Specific Interaction	605	Result of interaction.
XML-22 Relations		
Pharmacokinetic	706	Indicated by triggers, involves effect on absorption, distribution, metabolism and excretion of interacting drug.
Pharmacodynamic	1487	Indicated by triggers and specific Interactions, is the effect of the drug combination on the organism.
Unspecified	687	Indicated by triggers, are general warning of risk against combining label drug with precipitant.
NLM-180 Mentions		
Drug	4623	A drug is a chemical substance administered for medicinal purposes. Drug names can be generic (chemical name of the substance) or brand names .
Drug_Class	2800	Drug classes are assigned based on different characteristics: chemical ingredients, administration method, mechanism of action, and the target organ or target anatomical system.
Substance	191	Substances refer to any material entities that are not drug or drug classes. These include foods, nutritional supplements and other things that can be found in the environment.
Specific_Interaction	1870	Specific interactions indicate specific effects resulting from the interaction.
Increase_Interaction	810	Increase interactions indicate higher levels or increased effects of the object drug in the system as a result of the precipitant drug.
Decrease_Interaction	224	Decrease interactions indicate lower levels or decreased effects of the object drug in the system as a result of the precipitant drug.
Caution_Interaction	1107	Caution interactions are general precautions about the use of two entities together without specific mention of an effect.

Table 1: Data Description.

NLM-180	XML-22
Drug, DrugClass, Substance, Biomedical Entity	Precipitant
DecreaseInteraction, IncreaseInteraction	Trigger (Pharmacokinetic Interaction)
SpecificInteraction	Trigger + SpecificInteraction (Pharmacodynamic Interaction)
CautionInteraction	Trigger (Unspecified Interaction)

Table 2: mapping between NLM-180 and XML-22 DATASETS.

timestep t is computed by:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}^i * x_t + \mathbf{U}^i * h_{t-1} + \mathbf{b}^i) \\
\mathbf{f}_t &= \sigma(\mathbf{W}^f * \mathbf{x}_t + \mathbf{U}^f * h_{t-1} + \mathbf{b}^f) \\
\mathbf{o}_t &= \sigma(\mathbf{W}^o * \mathbf{x}_t + \mathbf{U}^o * h_{t-1} + \mathbf{b}^o) \\
\mathbf{g}_t &= \tanh(\mathbf{W}^g * \mathbf{x}_t + \mathbf{U}^g * h_{t-1} + \mathbf{b}^g) \\
\mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t \\
\mathbf{h}_t &= \mathbf{o}_t \odot \tanh(c_t)
\end{aligned} \tag{1}$$

where $\sigma(.)$ and $\tanh(.)$ are the element-wise sigmoid and hyperbolic tangent functions, $\odot$ is the element-wise multiplication operator, and i_t, f_t, o_t are the input, forget and output gates. h_{t-1}, c_{t-1} are the hidden state and memory cell of previous timestep respectively.

Unidirectional LSTMs suffer from the weakness of not utilizing the future contextual information. Bi-directional LSTM (Graves and Schmidhuber, 2005;

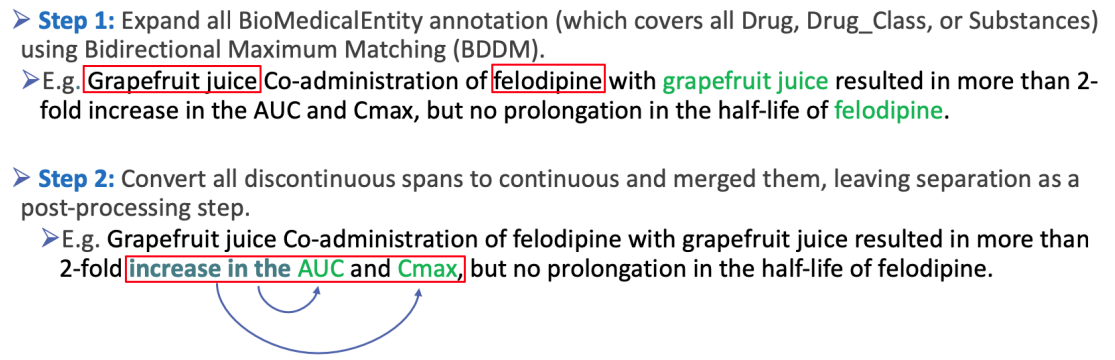

Figure 1: Example demonstrating preprocessing on NLM-180. Original annotations are shown in Green. Modified annotations are displayed in Red bounded box.

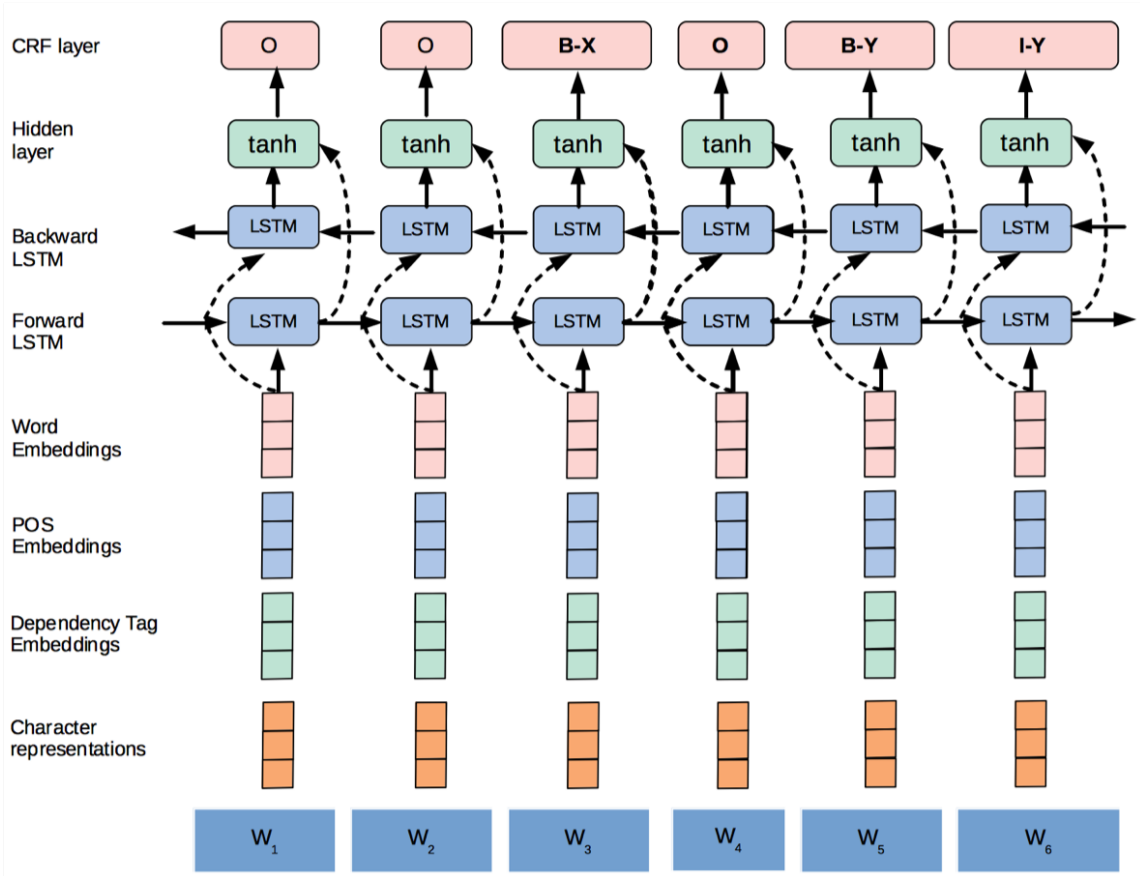

Figure 2: Concept Extraction.

Graves, 2013) addresses this by using two independent LSTMs (forward and backward) in which one processes the input sequence in the forward direction, while the other processes the input in the reverse direction. The forward LSTM computes the forward hidden states ($\overrightarrow{h_1}$, $\overrightarrow{h_2}$, ... $\overrightarrow{h_t}$) while the backward LSTM computes backward hidden states ($\overleftarrow{h_1}$, $\overleftarrow{h_2}$, ... $\overleftarrow{h_n}$) . Then for each timestep t , the hidden state of the Bi-LSTM is generated by concatenating $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$

$$\overleftrightarrow{h_t} = (\overrightarrow{h_t}, \overleftarrow{h_t}) \tag{2}$$

Although Bi-directional LSTM networks have the ability to capture long distance inter-dependencies, previous research suggests additionally capturing the correlations between adjacent labels can help in sequence labeling problems (Lample et al., 2016; Collobert et al., 2011; Huang et al., 2015). Conditional random fields (CRF) (Sutton et al., 2012) helps in capturing these correlations between adjacent tags. Given an observation sequence $\overleftrightarrow{h_t}$ (outputs from Bi-directional LSTM), CRF jointly models the probability of the entire sequence of labels Y=$(y_1, y_2 ... y_T)$ by using the discriminative probability to y_i given x_i and the transition probability between adjacent labels.

Using preprocessed NLM-180, dataset we trained a BiLSTM-CRF model to recognize all Biomedical Entities. We used sentence tokens, part-of-speech tags and dependency features as inputs to the model. Next, we trained another BiLSTM-CRF model using the same dataset to recognize all the Interaction (*Pharmacokinetic*, *Pharmacodynamic* and *Unspecified Interaction*) spans. We used sentence tokens, part-of-speech tags, dependency features and type features (BiomedicalEntity types) as inputs to the model.

3.2 Relation Extraction Architecture

Attention mechanism is a technique often used in neural translation of text introduced in (Bahdanau et al., 2014). The attention mechanism allows the networks to selectively focus on specific information. This has benefited several natural language processing (NLP) tasks such as factoid question answering (Hermann et al., 2015), machine translation (Bahdanau et al., 2014) and relation classification(Zhou et al., 2016). In this paper, we use attention mechanism for relation classification task (see Figure 3) similar to (Zhou et al., 2016).

Formally, let H be a matrix consisting of output vectors [$\overleftrightarrow{h_1}, \overleftrightarrow{h_2} .... \overleftrightarrow{h_t}$] (outputs from Bi-directional LSTM network), the representation r of the input is formed by a weighted sum of these output vectors:

$$\begin{aligned} \mathbf{M} &= \tanh(H) \\ \alpha &= softmax(w^T * M) \\ \mathbf{r} &= H * \alpha^T \end{aligned} \tag{3}$$

where $H \varepsilon R^{d^w XT}$, d^w is the dimention of vectors, w^T is the transpose of trained parameter vector. We obtain the final representation from:

$$\mathbf{h}^* = tanh(r) \tag{4}$$

4 Our System

4.1 Concept and Relation Extraction

We present our overall system architecture in Figure 4. As shown in Table 1, the number of annotations in XML-22 is much less compared to NLM-180, so, we used XML-22 only as development data and used NLM-180 to train our models. We split the preprocessed NLM-180 into 90% labels for our training set and the remaining 10% to tune model parameters. As a preprocessing step, we used Spacy (Honnibal and Montani) for tokenization, sentence segmentation, part-of-speech tagging and dependency parsing. We first trained two different concept extraction models (architecutre introduced in section) to recognize all Bio-medical Entities and Interaction spans. We used words, part-of-speech tags, dependency features, character representations as inputs to these model. We trained two separate models to avoid nested entities as Biomedical entities and Interaction spans sometimes overlap with each other in the dataset. Next, we trained relation extraction model (architecture introduced in section) that helps in determining the relation type (*hasObject, hasPrecipitant, NO_RELATION*) between recognized Biomedical Entities and Interactions. To train these models, We used words, part-of-speech tags,dependency features, type features and positional indicators as inputs to train this model. The word embeddings are pretrained using word2vec (Mikolov et al., 2013) on entirety of unannotated Structured Product Label (SPL) data. We fixed word embedding length to 200, character embedding length to 10, part-of-speech embedding length to 20 and dependency-parse to 20. Character, part-of-speech and dependency-parse embeddings are initialized randomly.

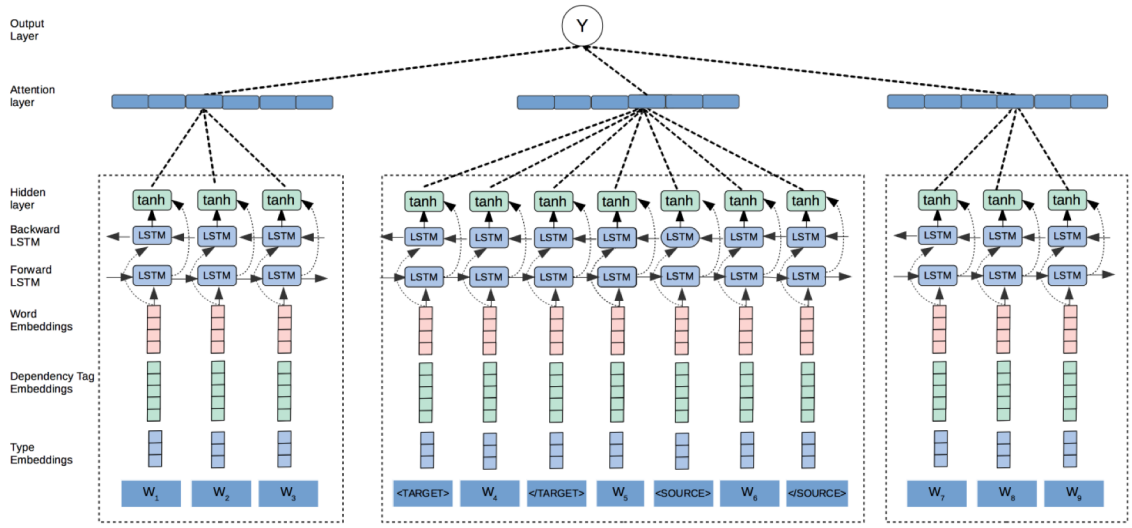

Figure 3: Architecture of Relation Extraction.

Using these trained models, we identified Biomedical Entities, Interactions and Relations in XML-22 datasets. Next, using XML-22 as development dataset, we developed a multi-step approach to map the identified concepts and relations to XML-22 annotations as shown below:

- we discarded: 1) all Biomedical entities that do not participate in a relation and 2) all biomedical entities that participate in hasObject relation if and only if the given biomedical entity belongs to same drug/brand named/drug class of the corresponding SPL.

- Next, we mapped the predicted concept types to the ones in XML-22 using the mapping described in Table 2 as illustrated in Figure 5

- Next, we developed a hybrid linguistic approach that combines shallow parsing, syntactic simplification with pattern matching to extract triggers from the recognized interactions and to further restore discontinuous spans as illustrated in Figure 6.

- Finally We performed clean up on each exrtracted concept at a token level, such as removal of certain POS tags, such as leading DET, PREP tags, as well as removal of any token predicted as a Biomedical Entity from Triggers and Specific Interaction entities. Others also involved removal of certain types of hedging words such as serious, life-threatening based on the DDI Guidelines.

parameter	BioMed	Interaction	RE
dropout	0.4	0.4	0.5
learning rate	0.02	0.03	0.03
reg.	$1e^{-7}$	$1e^{-6}$	$1e^{-6}$
hidden layer	150	100	100

Table 3: Hyperparameters for our system.

Hyperparameters There are four hyperparameters in our models, namely the dropout rate, learning rate, regularization parameter, and hidden layer size. The hyperparameters for our models were tuned on the development set for each task. Previous research suggests using dropout mitigates over-fitting and especially beneficial to the NER task(Ma and Hovy, 2016). We experimented by tuning the hyperparameters with different settings: dropout rates (0.0, 0.1, 0.2, 0.3 and 0.4,0.5), hidden layer sizes (100,150,200) and regularization parameter ($1e^{-5}, 1e^{-6}, 1e^{-7}$,$1e^{-8}$.). We chose Adam (Kingma and Ba, 2014) as our stochastic optimizer and tuned the learning rate at (0.01,0.02,0.03).We used early stopping(Graves, 2013) based on performance on development dataset. The best performance appear at around 20 epochs and 15 epochs for concept extraction and relation extraction respectively.

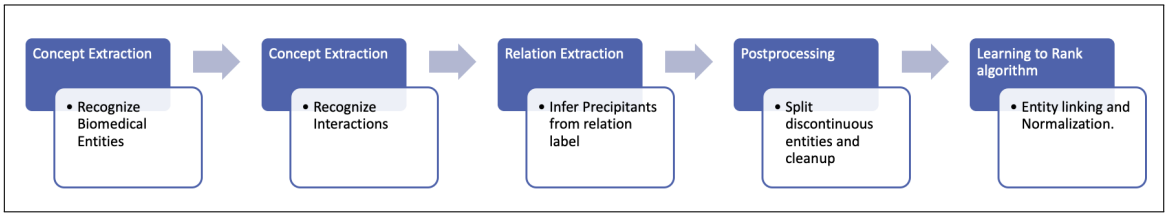

Figure 4: Our system.

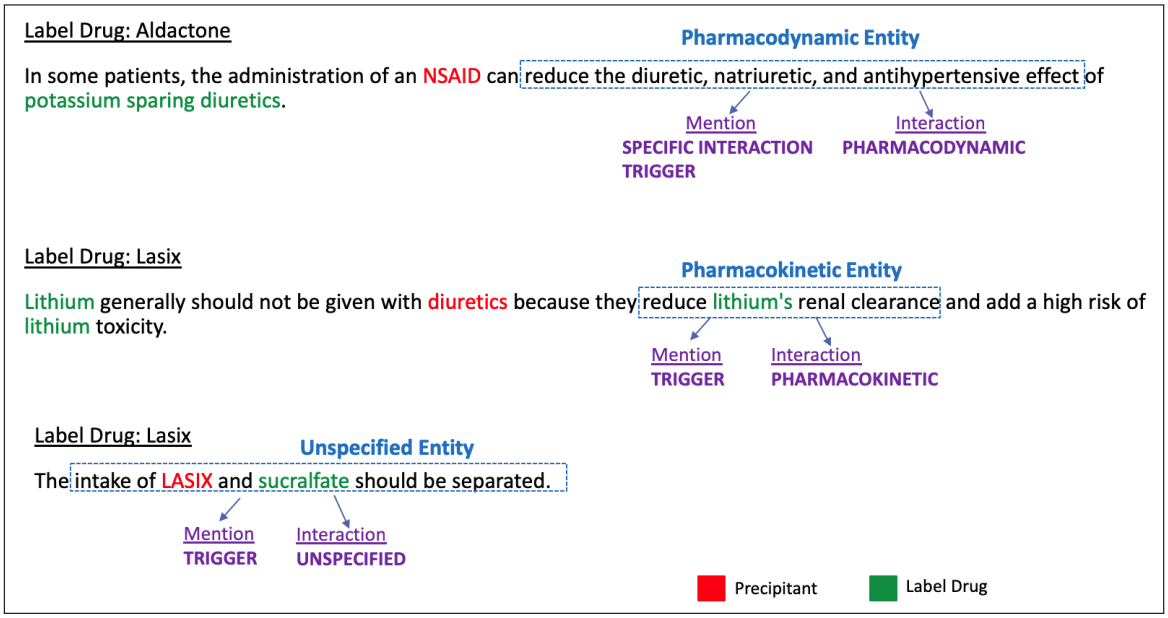

Figure 5: Illustration of Label Mapping from NLM-180 to XML-22 .

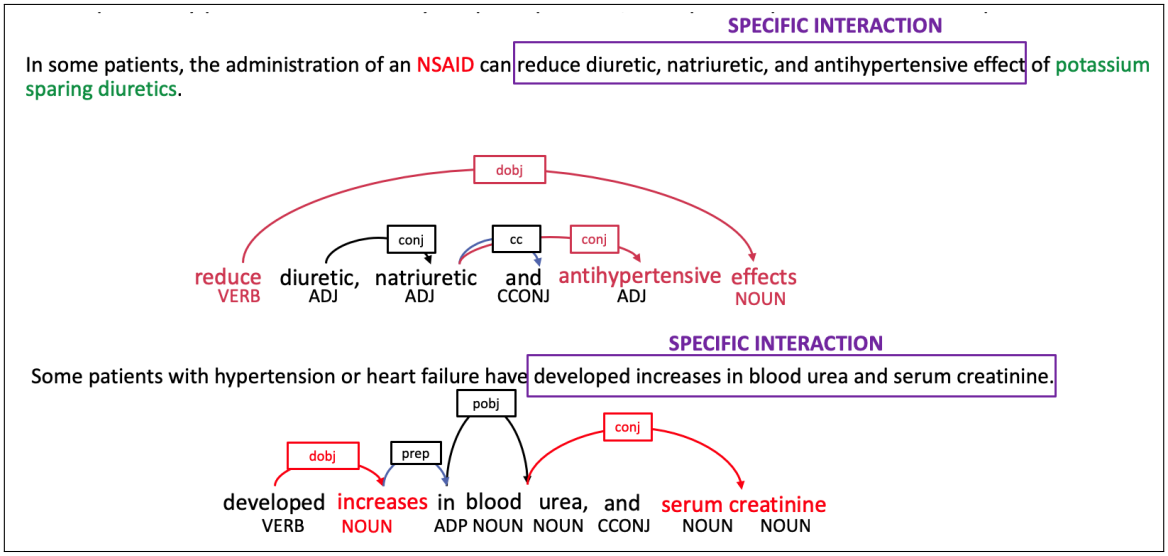

Figure 6: Illustration of extracting discontinuous spans.

4.2 Normalization

We use a learning to rank technique to perform the normalization task for specific interactions and precipitants. Formally, for a given mention m, we select the best term with the highest-ranking score from the corresponding knowledge source. we first employ the BM25 model provided by Lucene to retrieve the top 10 candidate terms for a given mention. Then, for each pair of a mention and a candidate term, we calculate four scores as matching features: BM25 ranking score, Jaccard similarity score, Longest common subsequence and word2vec similarity. Finally, we employ the linear RankSVM, one of the widely-used methods for learning to rank, to assign a final ranking score to each candidate term. The top ranked term for each mention is then chosen as the normalization for the mention. Finally, we developed simple heuristics based on the mention span and its associated relationships to match pharmacokinetic effects to National Cancer Institute Thesaurus codes.

4.3 Results

Concept Extraction			
	Precision	**Recall**	**F1**
Exact (-type)	42.21	23.55	30.06
Exact (+type)	41.94	23.19	29.87
Relation Extraction			
Binary	46.60	29.78	36.34
Binary (+type)	38.19	24.41	29.78
Full (-type)	25.24	16.10	19.66
Full (+type)	25.24	16.10	19.66
Normalization			
Micro	32.24	19.99	24.73
Macro	31.90	20.07	23.38
Distinct Interactions			
Micro	17.35	9.67	12.42
Macro	17.40	9.70	11.83

Table 4: Results for TEST1 DATASET.

Table 5 and Table **??** show our submitted results on test datasets for all tasks tasks. These results are obtained by using the hyperparameters shown in Table 3. These hyperparameters are obtained by tuning them on development set. The fact that F1-score dropped from Task-1 to Task-4 by a huge margin indicates interactions that are repeated in multiple sections are captured easily when compared with interactions that are mentioned less number of times. Furthermore, extracting triggers and effects of the interactions are harder compared to precipitants. Thus,

Concept Extraction			
	Precision	**Recall**	**F1**
Exact (-type)	44.96	29.45	35.59
Exact (+type)	44.61	29.31	35.38
Relation Extraction			
Binary	50.07	36.86	42.46
Binary (+type)	40.77	30.02	34.58
Full (-type)	22.99	16.83	19.43
Full (+type)	22.99	16.83	19.43
Normalization			
Micro	30.82	26.28	28.37
Macro	26.46	23.90	24.52
Distinct Interactions			
Micro	12.66	9.59	10.91
Macro	9.74	7.79	8.45

Table 5: Results for TEST2 DATASET.

The results on Task 4 clearly indicate producing index files coded to multiple terminologies fully automatically is unattainable at this time. However, the availablility of datasets, especially considering the size of training and test datasets combined, is encouraging and will promote further research to address this important and challenging problem.

5 Conclusion and Future Work

We reported on using state-of-the-art deep learning neural networks for identifying mentions and relations relevant to DDI extraction. In this research, we proposed a methodology to efficiently map NLM-180 dataset to official 22 drug labels. Furthermore, we are the only team to demonstrate a complete end-to-end system to extraction drug-drug interactions. Although, the overall results on official test sets are not very encouraging, this results will inform future FDA efforts at automating important safety processes, and could potentially lead to future FDA collaboration with interested researchers in this area. Our future directions include: Potential headroom for improvements:

- Incorporate Knowledge bases (such as UMLS) into deep learning models to accurately identify precipitants and interactions.
- Hybrid approach that can better leverage syntactic dependencies to break co-ordinate conjunctions.
- Measure the impact of deep learning models with respect to size of the training data i.e., us-

ing the official test set (123 XML documents) to train the models and evaluate on XML-22.

- Leveraging complex encodings such as BIOHD to encode discontinuous spans into the model.

Leveraging complex encodings such as BIOHD to encode discontinuous spans into the model.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* .

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks* 5(2):157–166.

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

Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850* .

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks* 18(5):602–610.

Karl Moritz Hermann, Tomáš Kociskỳ, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *arXiv preprint arXiv:1506.03340* .

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

Matthew Honnibal and Ines Montani. ???? spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear* .

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

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360* .

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354* .

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* .

Charles Sutton, Andrew McCallum, et al. 2012. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning* 4(4):267–373.

Dongxu Zhang and Dong Wang. 2015. Relation classification via recurrent neural network. *arXiv preprint arXiv:1508.01006* .

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. volume 2, pages 207–212.