Proceedings of Machine Learning Research 90:39–47, 2018 Medication and Adverse Drug Event Detection Workshop

IBM Research System at MADE 2018: Detecting Adverse Drug Events from Electronic Health Records

Bharath Dandala BDAND@US.IBM.COM
Venkata Joopudi VNJOOPUD@US.IBM.COM
Murthy Devarakonda* MURTHY.DEVARAKONDA@ASU.EDU
IBM Research, Yorktown Heights, NY.

Editor: Feifan Liu, Abhyuday Jagannatha, Hong Yu

Abstract

Adverse Drug Events (ADEs) are common and occur in approximately 2-5% of hospitalized adult patients. Each ADE is estimated to increase healthcare cost by more than $3,200. Severe ADEs rank among the top 5 or 6 leading causes of death in the United States. Prevention, early detection and mitigation of ADEs could save both lives and dollars. Employing Natural Language Processing (NLP) techniques on Electronic Health Records (EHRs) provides an effective way of real-time pharmacovigilance and drug safety surveillance. Thus, in this research, we developed a system for three different NLP tasks namely: Named Entity Recognition (NER), Relation Identification and Integrated task (integrative system to conduct NER and relation identification together). Our system achieved F-1 measures of 0.829 for Named Entity Recognition, 0.840 for Relation Identification and 0.617 for Integrated task. Our system ranked 1st in the integrated task and 2nd in both entity extraction and relation identification tasks.

Keywords: Deep Learning, Adverse Drug Events, Named Entity Extraction, Relation Identification, BiLSTM-CRF, Attention-BiLSTM

1. Introduction

Information extraction methods for named entity recognition (NER) and relation identification is a fundamental requirement in automatic adverse drug event extraction. Accuracy of these foundational analytics will significantly impact adverse drug reaction curation and further, has the potential to improve clinical decision support systems. BiLSTM-CRF models (Huang et al., 2015) have previously shown to accurately recognize entities in biomedical and clinical corpora (Chalapathy et al., 2016; Habibi et al., 2017; Li et al., 2017; Dandala et al., 2017). In this research, we used BiLSTM-CRF models for named entity recognition. 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, which has benefited serveral 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 used Attention mechanism for relation classification task similar to Zhou et al. (2016).

* Murthy Devarakonda is now at Arizona State University.

The rest of the paper is organized as follows: Section 2 describes the dataset; Section 3 discusses our system architecture for the entity and relation identification tasks; Section 4 describes our methods and the experimental settings of our system and in Section 5, we conclude with insights and future work.

2. Datasets and pre-processing

The entire dataset contains 1092 de-identified clinical notes from Electronic Health Records (EHRs) of 21 cancer patients. Each EHR note was annotated with medication information (name, dosage, route, frequency, duration), adverse drug events (ADEs), indications, other signs and symptoms (SSLIFs), and relations among those entities. An SSLIF is labeled as an ADE if it is a *side effect* of a drug and it is labeled as *Indication* if it is an affliction that a doctor is actively treating with a medication. An important characteristic of ADEs or Indications is that their labels (ADE, Indication) are not only determined by intra-sentential but also by inter-sentential contexts. In this dataset, only 61% of the ADEs and 46% of Indications participate in "adverse" or "reason" relationship within the same sentence respectively. Thus, it is important to capture inter-sentential relationships between entities to associate them with appropriate label. Jagannatha and Yu (2016) demonstrated the importance of inter-sentential contexts to improve NER system by capturing the sequential information at the document level (LSTM-document) rather than at the sentence level.

In this research, we used sentences as logical units of contextual information. In entity extraction task, contextual evidence in the sentence is used to identify the span of the entity and associate with its corresponding type. However, In relation identification task, for a given pair of entities in a document, sentences in which they appear as well as sentences that are between them in the original document serve as contextual evidence in determining the relation type. Thus, it is important to obtain accurate sentence segments as the context for a sequence model is limited to the words present in the sentence. However, sentence segmentation is a non-trivial task in clinical notes. Unlike regular text passages, sentences in a note do not always end with regular punctuation marks and new line characters are introduced by textwrap settings in EHR systems. Manually Inspecting this dataset, we determined the textwrap is around 80 characters.

As a first step, We identified logical blocks of text which we refer as *pseudo-paragraphs*. To identify the pseudo-paragraphs, we first selectively replaced the newline characters with white space characters. For each line of text, we replaced the newline character with a white space, if the length of the line was between 70 and 85 (because textwrap is around 80). As a next step, we split the lines to multiple blocks or pseudo-paragraphs considering given two new-line characters occur consecutively. The identified blocks or pseudo-paragraphs are then fed into Stanford sentence segmenter for identifying sentences within that pseudo-paragraph. Furthermore, we used Stanford parser Manning et al. (2014) for tokenization, sentence segmentation and parts-of-speech tagging.

3. Entity and Relation Extraction

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 bi-directional LSTMs have been proven to be effective (Huang et al., 2015; Ma and Hovy, 2016; Zhou et al., 2016; Zhang and Wang, 2015). In this section, we describe our NER and relation identification systems in detail.

3.1. Entity Extraction

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 problem (Bengio et al., 1994) of vanilla RNNs by using adaptive gating mechanism. 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. Thus, In this research, we used BiLSTM-CRF for entity extraction similar to Huang et al. (2015).

As mentioned earlier, only 61% of the ADEs and 46% of indications occur in "adverse" or "reason" relationship within the same sentence. This implies that the required context for the remaining instances is present across multiple sentences. For these two types, any model that only takes the context within a sentence will not be sufficient. Hence, we perform our entity extraction over two steps. In the first step, we used a BiLSTM-CRF neural network to model generic entity types. Generic entity types are obtained by replacing ADE and Indication labels with SSLIF label from the original training data. In the second step, the predictions from the relation identification task are used to infer the original type from the generic type. For example, the target entity of an "adverse" relation is updated to ADE type (from SSLIF). Character embeddings, word embeddings and part-of-speech embeddings are provided as inputs to the BiLSTM-CRF network.

3.2. Relation Identification

We used Attention-BiLSTM architecture introduced by Zhou et al. (2016) for relation identification. This network takes tokens, types (outputs of entity extraction model) and positional indicators of source and target concepts as inputs. As introduced in section 2, this dataset contains intra and inter-sentential relationships. The entities participating in an inter-sentential relation can occur anywhere in a document; thus resulting a large number of possible entity-pairs. While it is trivial to take all entity pairs in a EHR note to achieve 100% recall, this results in a highly unbalanced dataset containing a large number of negative relation instances. Previous research on inter-sentential relation extraction (Swampillai and Stevenson, 2011; Quirk and Poon, 2016; Peng et al., 2017) suggests addressing this issue either by undersampling the negative class or by training a cost-sensitive classifier helps in learning better relation extraction model. Inspired by this, as a preprocessing step, we selectively undersampled negative examples by using the following heuristics:

- For each relation type, we estimated the relative number of sentences between source and target entities using the training data. We used this heuristic to control/reduce the number of negative examples. For example, forming relation pairs between all Medications and SSLIFs (indications/ADEs) that are within 7 sentences (in both

positive and negative direction) yields 98% recall. While, we lost 2% recall, we significantly reduced (about 70%) the number of negative Medication-SSLIF pairs. Similarly, for manner/route relation type, forming relation pairs. between medication and route entities (within 1 sentence in both forward and backward directions) yields 99% recall. This heuristic is extremely useful in removing a large number of negative examples.

- Additionally, we developed a rule-based/heuristic-based methodology to identify section boundaries. We used section labels as boundaries to form relation pairs that occur only within a given section. This further helped in removing the negative pairs without losing much recall.
- Each relation type has a dominant pair of source and target entity types. For example, majority of dosage relation instances have medication as source entity and dose as target entity. For each pair of source and target entity types, we considered only dominant type of relation as valid and removed the pairs with other relation labels.

3.3. Task specific embeddings

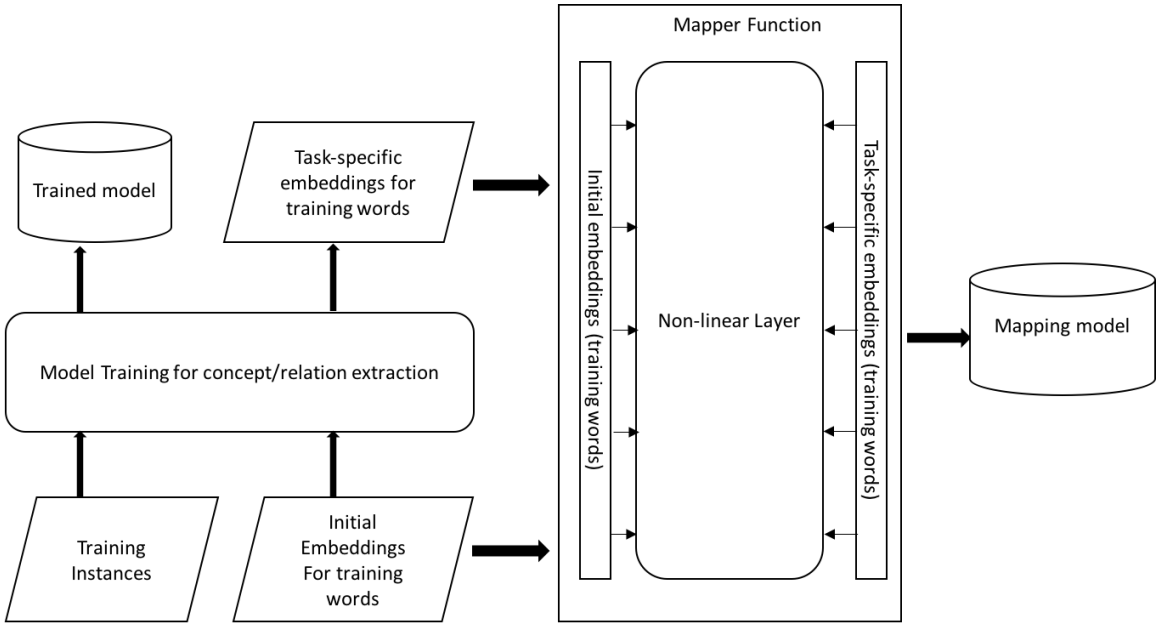

Figure 1: Mapper training for task-specific embeddings.

Performance of NLP tasks drops significantly when moving from training data to held-out dataset (Petrov et al., 2010). One of the primary reasons for such a drop is words that do not appear in the training data appear in test data (referred as out-of-training vocabulary). In deep learning architectures, it is a common practice to learn one embedding for all the rare words in training data and using this learned representation for all unseen words during the testing. Another alternative is to use intial embeddings (pre-trained embeddings

obtained from large dataset) for all unseen words. However, such an approach often leads to errors (Madhyastha et al., 2015).

Recent research suggests (Madhyastha et al., 2015; Jhamtani et al., 2017) learning task-specific embeddings for unseen words from a large vocabulary helps in improving NLP tasks. This methodology has significant effect especially when dealing with data such as EHRs because: 1) The total number of variations, for example medication name SSLIFs), in training data are very small compared to the real world data, and 2) sentences are short or often times they occur without any contextual information. Learning task-specific embeddings can significantly help such cases by leveraging embeddings from external resources and adapting them to the task-at-hand.

As shown in Figure 1, we learn a non-linear mapper function to map from initial embeddings to the task-specific embeddings space, via a multi-loss objective function using words in the training data. Finally, the learned mapping function is used to transform the initial embeddings of all out-of-training words and we use these transformed embeddings to represent the unseen words during the test phase. In our system, task-specific embeddings are learnt for both entity extraction and relation identification models.

4. Experimental Settings and Results

We divided the released training dataset into two additional datasets (development and an internal test dataset). Development data was 10% and internal test data was 20% of the released training data. We used the word embeddings that are released as part of the challenge (Jagannatha and Yu, 2016). We fixed word embeddings size to 200, character embeddings size to 50 and part-of-speech embeddings length to 20. The part-of-speech and character embeddings are initialized randomly.

4.1. Hyperparameter tuning

There are four hyper-parameters 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, 0.4 and 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 ($1e^{-2}, 1e^{-3}, 1e^{-4}$). 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 and relation extraction respectively. We performed hyper-parameter tuning and used the network parameters listed in 1. We used both dropout and L2 regularization for optimizing the network parameters.

4.2. Results

Table 2 and Table 3 shows our results on internal and released test datasets for the entity and relation extraction tasks respectively. These results are obtained by using the

Table 1: Network hyperparmeters

Parameter	**Entity Extraction**	**Relation Identification**
Dropout	0.5	0.5
Learning rate	1e-3	1e-4
Regularization	1e-6	1e-4
Hidden layer size	200	100

hyperparameters shown in Table 1. As a first experiment, we used BiLSTM-CRF for concept extraction and Attention-BLSTM for relation extraction. In this experiment, during the evaluation phase, we used pre-trained embeddings for unseen words. We refer to this experiment as Model-A. To understand the importance of task-specific embeddings for unseen/rare words, we conducted a separate experiment in which we learned task-specific embedding for all unseen words during the training phase as explained in Section 3.3. We refer to this experiment as Model-B. Table 2 compares the performance of these experiments on our internal test dataset.Table-3 lists the performance of Model-B on the external test dataset.

Table 2: Performance on internal test dataset

Task	**Model-A**			**Model-B**		
	Precision	Recall	F-1	Precision	Recall	F1
Entity Extraction	0.889	0.860	0.874	0.893	0.885	0.889
Relation Identification	0.954	0.921	0.937	0.955	0.925	0.940
Integrated task	0.833	0.703	0.762	0.845	0.719	0.777

Table 3: Performance on external test dataset

Task	**F1**
Entity Extraction	0.8285
Relation Identification	0.8402
Integrated task	0.6170

On the internal test dataset, Model-A achieved F-measures of 0.874, 0.937 and 0.762 for the entity extraction, relation extraction and integrated task respectively. As shown in Table 2, using task-specific embeddings for unseen words achieved better performance for all the three tasks. For a fair comparision, we used the same parameter in all these systems and used same input features/embeddings.

Table 3 shows our results on the external dataset released by the organizers. Among all the submitted systems to MADE, our systems achieved second-best accuracy in both entity extraction and relation identification tasks and achieved the highest accuracy overall. Our system significantly outperforrmed the other systems in the integrated task.

5. Conclusions and Future Work

Here, we reported on using state-of-the-art deep learning neural networks for identifying entities and relations relevant to ADEs. We used a two-stage process for entity extraction by using both BiLSTM-CRF and Attention-BiLSTM models. We achieved state-of-art results on entity and relation extraction tasks and developed the best overall system for the integrated task. Furthermore, higher accuracy of Model-B over Model-A highlights the importance of learning task specific embeddings for unseen words. The following section lists the future steps that can be explored:

- Embeddings - Using embeddings trained on a large number of Electronic Health Records. Especially, this may benefit identifying concepts which are spelling mistakes or abbreviations.
- Representation - Handling nested concepts (about 1%) i.e, (span of one or more concepts overlaps with each other) with more advanced decoding schemes that avoid lossy representation.
- Semi-structured nature of EHRs - EHR data has rich semi-structured information such as sub-headings and structured contents (medication, diagnosis tables). In the future, we plan to build cascading models which can also exploit this structured information as well as semi-structured information in more depth.
- External knowledge - Incorporating external knowledge from manually curated semantic networks such as UMLS or incorporating associations extracted from large number of EHRs to identify relationships. This is particularly helpful in identifying relations which do not have sufficient lexical cues between the source and target entities.

References

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

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

Raghavendra Chalapathy, Ehsan Zare Borzeshi, and Massimo Piccardi. Bidirectional lstm-crf for clinical concept extraction. *arXiv preprint arXiv:1611.08373*, 2016.

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(Aug):2493–2537, 2011.

Bharath Dandala, Mahajan Diwakar, and Devarakonda Murthy. Ibm research system at tac 2017: Adverse drug reactions extraction from drug labels. In *Text Analysis Conference (TAC2017)*. NIST, 2017.

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

Maryam Habibi, Leon Weber, Mariana Neves, David Luis Wiegandt, and Ulf Leser. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48, 2017.

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

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

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

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

Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. Shakespearizing modern language using copy-enriched sequence-to-sequence models. *arXiv preprint arXiv:1707.01161*, 2017.

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

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

Fei Li, Meishan Zhang, Bo Tian, Bo Chen, Guohong Fu, and Donghong Ji. Recognizing irregular entities in biomedical text via deep neural networks. *Pattern Recognition Letters*, 2017.

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

Pranava Swaroop Madhyastha, Mohit Bansal, Kevin Gimpel, and Karen Livescu. Mapping unseen words to task-trained embedding spaces. *arXiv preprint arXiv:1510.02387*, 2015.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, et al. The stanford corenlp natural language processing toolkit. In *In ACL, System Demonstrations*. Citeseer, 2014.

Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen-tau Yih. Cross-sentence n-ary relation extraction with graph lstms. *arXiv preprint arXiv:1708.03743*, 2017.

Slav Petrov, Pi-Chuan Chang, Michael Ringgaard, and Hiyan Alshawi. Uptraining for accurate deterministic question parsing. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 705–713. Association for Computational Linguistics, 2010.

Chris Quirk and Hoifung Poon. Distant supervision for relation extraction beyond the sentence boundary. *arXiv preprint arXiv:1609.04873*, 2016.

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

Kumutha Swampillai and Mark Stevenson. Extracting relations within and across sentences. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 25–32, 2011.

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

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 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, 2016.