The Dots Have Their Values: Exploiting the Node-Edge Connections in Graph-based Neural Models for Document-level Relation Extraction

Hieu Minh Tran[1], **Minh Trung Nguyen**[1] **and Thien Huu Nguyen**[2]
[1] VinAI Research, Vietnam
[2] Department of Computer and Information Science, University of Oregon,
Eugene, OR 97403, USA
{v.hieutm4,v.trungnm11}@vinai.io, thien@cs.uoregon.edu

Abstract

The goal of Document-level Relation Extraction (DRE) is to recognize the relations between entity mentions that can span beyond sentence boundary. The current state-of-the-art method for this problem has involved the graph-based edge-oriented model where the entity mentions, entities, and sentences in the documents are used as the nodes of the document graphs for representation learning. However, this model does not capture the representations for the nodes in the graphs, thus preventing it from effectively encoding the specific and relevant information of the nodes for DRE. To address this issue, we propose to explicitly compute the representations for the nodes in the graph-based edge-oriented model for DRE. These node representations allow us to introduce two novel representation regularization mechanisms to improve the representation vectors for DRE. The experiments show that our model achieves state-of-the-art performance on two benchmark datasets.

1 Introduction

An important task of Information Extraction is Relation Extraction (RE) that seeks to identify the semantic relationships between entities mentioned in text. The prior works have mainly focused on the intra-sentence scenario where the two entity mentions appear in the same sentences (Zhou et al., 2005; Zeng et al., 2014; Nguyen and Grishman, 2015). In this work, we study a more recent setting for RE that additionally considers relations between two entity mentions in different sentences of the documents (i.e., inter-sentence relations) (called document-level RE (DRE)).

The current methods for document-level RE have intensively relied on deep learning to induce effective representation vectors for relation prediction. Among these deep learning models, graph-based neural networks have been demonstrated as one of the most effective approaches for DRE due to their ability to capture long-distance and inter-sentential information in text (Peng et al., 2017; Quirk and Poon, 2017; Gupta et al., 2019). In particular, (Christopoulou et al., 2019) has recently introduced a graph-based edge-oriented network that achieves state-of-the-art performance for DRE. The key idea in this model is to build a interaction graph for each input document where the nodes include the entity mentions, the entities, and the sentences. Note that this is fundamentally different from the prior graph-based models for RE that have mostly used words as the nodes for the graphs (Zhang et al., 2018; Gupta et al., 2019). In this model, the edges between these nodes are determined by the coreferences of the entity mentions and the appearance of the entity mentions in the sentences. The representation vectors for the edges of the graph (thus called edge-oriented) are then computed via several inference layers, serving as the features to predict the relations between the pairs of entities in the documents. In this way, the model can leverage the interactions between the nodes and edges of different types to obtain richer representation vectors for the edges between the entity nodes (Christopoulou et al., 2019).

Despite its good performance, a major limitation of the graph-based edge-oriented model for DRE is that it only focuses on the edge representations and ignores the representations for the nodes of the graphs. On the one hand, this edge-only representation approach cannot explicitly encode the information that is specific to the entities/entity mentions in the documents (as it only captures the representations of pairs of entities/entity mentions), potentially missing an important clue to boost the performance for DRE (e.g., entity subtypes). One the other hand, the lack of the node representations prevents the models from exploiting the relations between the node and edge representations (e.g.,

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 4561–4567
November 16 - 20, 2020. ©2020 Association for Computational Linguistics

the translation relation (Bordes et al., 2013)) and the similarity for the representation vectors of the entity mention nodes (i.e., of the same entities) to enhance the representation learning for DRE. Consequently, in this work, we propose to improve the graph-based edge-oriented model for DRE by explicitly computing the representations for the nodes in the graphs. Based on such node representations, we introduce two novel regularization techniques to improve the representations for DRP, capturing the similarities between the node and edge representations of the same edges or the same entities. We conduct extensive experiments to demonstrate the effectiveness of the proposed method for DRE, yielding the state-of-the-art performance for this task on two benchmark datasets.

2 Related Work

RE has been extensively studied in the inter-sentence (Zelenko et al., 2003; Zhou et al., 2005; Zeng et al., 2014; Nguyen and Grishman, 2014, 2016; Zhang et al., 2018; Veyseh et al., 2019, 2020) and distant supervision settings (Mintz et al., 2009; Riedel et al., 2010; Zeng et al., 2015; Lin et al., 2016; Jiang et al., 2016; Zeng et al., 2017; Vashishth et al., 2018). Recently, document-level RE has gained more attention from the community. Two major approaches have been considered for DRE, i.e., the graph-based (Quirk and Poon, 2017; Peng et al., 2017; Song et al., 2018; Gupta et al., 2019; Jia et al., 2019; Sahu et al., 2019) and non-graph-based (Gu et al., 2017; Peng et al., 2016; Zhou et al., 2016; Zheng et al., 2018; Li et al., 2018; Verga et al., 2018; Nguyen and Verspoor, 2018; Ye et al., 2019; Singh and Bhatia, 2019) approaches. The closest work to ours is the graph-based edge-oriented model in (Christopoulou et al., 2019) that introduces new document graphs for DRE based on entity mentions, entities and sentences as the nodes.

3 Model

Formally, in the DRE problem, the input involves a document D with S, M, and E as the sets of the sentences, entity mentions, and entities (respectively) in D. The goal of DRE is to predict the semantic relationships between each pair of entities in E (i.e., including the type *NONE* for the entity pairs with no relations). In this section, we will first describe the graph-based edge-oriented model (EoG) in (Christopoulou et al., 2019). Afterward, we will present our novel node representation computation and regularization for this model.

3.1 The Graph-based Edge-oriented Model

In EoG, the words in the sentences in S are first transformed into vectors using some pre-trained word embeddings. A bidirectional Long-short Term Memory (BiLSTM) network is then applied over the sentences in S (i.e., treated as word embedding vector sequences) to obtain contextualized representation vectors for the words in D (called the BiLSTM vectors for the words). Afterward, EoG constructs an interaction graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to compute the representations for the entity pairs in E for relation prediction. In particular, the node set $\mathcal{V}$ in $\mathcal{G}$ involves three types of nodes: the sentence nodes n_s for each sentence s in S, the entity mention nodes n_m for each entity mention m in M, and the entity nodes n_e for each entity e in E. In EoG, each node in $\mathcal{V}$ is associated with an initial embedding vector to facilitate the edge representation computation later. The embedding vectors $\mathbf{n}_s$, $\mathbf{n}_m$ and $\mathbf{n}_e$ for the sentence node n_s, the mention node n_m and the entity node n_e (respectively) are formed by: $\mathbf{n}_s = [\text{avg}_{w \in s}\mathbf{w}, \mathbf{t}_s]$, $\mathbf{n}_m = [\text{avg}_{w \in m}\mathbf{w}, \mathbf{t}_m]$ and $\mathbf{n}_e = [\text{avg}_{m \in e}\mathbf{n}_m, \mathbf{t}_e]$ where: avg is the averaging operation of the vectors in a set, $\mathbf{w}$ is the BiLSTM vector of the word w, $[]$ is the vector concatenation, and $\mathbf{t}_s$, $\mathbf{t}_m$, and $\mathbf{t}_e$ are the embedding vectors to specify the types of the nodes.

Given the nodes $\mathcal{V}$, the edges in $\mathcal{E}$ in EoG are non-directed and include the five major types, i.e., Mention-Mention, Mention-Sentence, Mention-Entity, Sentence-Sentence, and Entity-Sentence. These edge types essentially employ the coreference information of the entity mentions, their correspondence with the entities and their appearance in the sentences to establish the edges in $\mathcal{E}$ (Christopoulou et al., 2019).

In EoG, each edge $z = (i, j) \in \mathcal{E}$ would be assigned with an initial embedding vector $\mathbf{e}^1_z$ based on the initial embedding vectors for the nodes $\mathbf{n}_i$ and $\mathbf{n}_j$ described above (Christopoulou et al., 2019). These initial edge embedding vectors will then be fed into N inference layers to produce the representation vectors for the entity pairs in E for relation prediction. In particular, in the l-th inference layer ($1 \leq l \leq N$), EoG computes the edge embedding vector $\mathbf{e}^{2^l}_{ij}$ for two nodes $i \neq j \in \mathcal{N}$ by:

$$\mathbf{e}^{2^l}_{(i,j)} = \beta \mathbf{e}^{2^{l-1}}_{(i,j)} + (1-\beta) \sum_{k \neq i,j} \sigma(\mathbf{e}^{2^{l-1}}_{(i,k)} \odot (W_E \mathbf{e}^{2^{l-1}}_{(k,j)})) \quad (1)$$

where σ is the sigmoid function, $\odot$ is the element-wise product, and $\beta \in [0, 1]$ is a controlling constant. Note that the representation vector $\mathbf{e}^{2^l}_{(i,j)}$ is able to capture paths with the length up to 2^l in $\mathcal{G}$. The representation vectors for the entity node pairs in the last inference layer (i.e., $\mathbf{e}^{2^N}_{(e_i,e_j)}$) would be used to perform relation prediction in EoG.

3.2 The Proposed Model

The Node Representation: As mentioned in the introduction, a problem with the original EoG model is the failure to exploit the representations for the nodes in $\mathcal{N}$ to improve the representations for DRE. In this work, we propose to explicitly compute the node representation vectors and use them to aid the representation learning for DRE. In particular, starting with the initial embedding vectors for the nodes $\mathbf{n}^1_s$, $\mathbf{n}^1_m$, and $\mathbf{n}^1_e$ (i.e., $\mathbf{n}^1_s = \mathbf{n}_s$, $\mathbf{n}^1_m = \mathbf{n}_m$, and $\mathbf{n}^1_e = \mathbf{n}_e$ for the nodes in $\mathcal{N}$), we obtain the representation for the node $i \in \mathcal{N}$ at the l-th inference layer via (motivated by the edge representation computation in EoG) ($1 \leq l \leq N$):

$$\mathbf{n}^{2^l}_i = \gamma \mathbf{n}^{2^{l-1}}_i + (1-\gamma) \sum_{(i,j)\in\mathcal{E}} \sigma(\mathbf{n}^{2^{l-1}}_i \odot (W_N \mathbf{n}^{2^{l-1}}_j)) \quad (2)$$

where γ is a controlling factor. With these node representation vectors, we predict the relation between the entities $e_i, e_j \in E$ by first forming a feature vector $V_{e_i,e_j} = [\mathbf{n}^{2^N}_{e_i}, \mathbf{n}^{2^N}_{e_j}, \mathbf{e}^{2^N}_{(e_i,e_j)}]$ (i.e., the vectors in the last inference layers). V_{e_i,e_j} would then be sent to a feed-forward network with the softmax layer in the end to produce the distribution $P_{e_i,e_j}(.|e_i, e_j)$ over the possible relations for DRE. Finally, the negative log-likelihood $\mathcal{L}_{pred}$ for all the entity pairs in E would be employed to train the models in this work:

$\mathcal{L}_{pred} = -\sum_{e_i \neq e_j \in E} \log P_{e_i,e_j}(y_{e_i,e_j}|e_i, e_j)$
where y_{e_i,e_j} is the golden relation for e_i and e_j.

The Node-Edge Representation Consistency: The introduction of the node representation vectors allows us to leverage the relations between the representation vectors of an edge $(i, j) \in \mathcal{V}$ and its two end nodes (i.e., $i, j \in \mathcal{N}$) to regularize the representations, potentially improving their quality for DRE. In particular, motivated from the knowledge graph embedding methods (Bordes et al., 2013), in this work, we propose to enforce the representation vectors for the nodes and edges in $\mathcal{G}$ to follow the translation relation (i.e., the sum of the representation for the node $i \in \mathcal{N}$ and the representation for the edge $(i, j) \in \mathcal{E}$ should be as close as possible to the representation for the node $j \in \mathcal{N}$). Formally, this amounts to adding the following term $\mathcal{L}_{rel}$ into the overall loss function to train the models: $\mathcal{L}_{rel} = \sum_{l=1}^{N} \sum_{(i,j)\in\mathcal{E}} ||\mathbf{n}^{2^l}_i + \mathbf{e}^{2^l}_{(ij)} - \mathbf{n}^{2^l}_j||$.

The Mention Representation Consistency: In addition to the node-edge consistency, the explicit representations for the nodes in $\mathcal{G}$ facilitates the use of the coreference information between the entity mentions to constrain the representations for DRE. Specifically, to further improve the representation vectors for DRE, we propose to encourage the embedding vectors for the entity mention nodes of the same entities to be similar to each other. This is based on the intuitive assumption that the embedding vectors for the coreferred entity mentions should capture the underlying semantics/information of the entity they are referring to, thus being close to each other. We expect that this explicit similarity regulation between the entity mention representations would help to regularize the embedding vectors to encode more meaningful information for DRE. In particular, to achieve such similarities, we propose to incorporate the following loss term $\mathcal{L}_{const}$ into the overall loss function:

$\mathcal{L}_{const} = \sum_{l=1}^{N} \sum_{e\in\mathcal{E}} \sum_{m_i \neq m_j \in e} \sum_{l=1}^{N} (1 - cosine(\mathbf{n}^{2^l}_{m_i}, \mathbf{n}^{2^l}_{m_j}))$.

Here, $1 - cosine(\mathbf{n}^{2^l}_{m_i}, \mathbf{n}^{2^l}_{m_j})$ is to measure the similarity between $\mathbf{n}^{2^l}_{m_i}$ and $\mathbf{n}^{2^l}_{m_j}$.

To summarize, the overall loss function $\mathcal{L}$ in this work is: $\mathcal{L} = \mathcal{L}_{pred} + \alpha_{rel}\mathcal{L}_{rel} + \alpha_{const}\mathcal{L}_{const}$ with α_{rel} and α_{const} as the trade-off parameters. The EoG model Augmented with Node Representations in this work is called EoGANE for convenience.

4 Experiments

Datasets & Parameters: We evaluate the models on two benchmark datasets for DRE, i.e., **CDR** and **GDA**. The CDR (Chemical-Disease Reactions) dataset is manually annotated for the binary interactions between Chemical and Disease concepts (Li et al., 2016a) while GDA (Gene-Disease Associations) (Ye et al., 2019) provides the annotations for the binary interactions between Gene and Disease concepts using distant supervision. For both datasets, we follow the same data preprocessing and spits (i.e., for training/development/test data) as the prior work (Christopoulou et al., 2019) to achieve a fair comparison. Also, similar to (Christopoulou et al., 2019), we use the PubMed pre-trained word embeddings (Chiu et al., 2016) for the models on CDR while randomly initialized word embeddings are employed for GDA. These word embeddings are optimized during the training process of the models in this work.

We implement the EoGANE model in this work by extending the code for the EoG model that is provided in its original paper (Christopoulou et al., 2019). As such, we inherit the values for the common hyper-parameters between EoG and EoGANE from (Christopoulou et al., 2019) for the proposed model EoGANE (e.g., 0.002 for the learning rate of the Adam optimizer, 2 and 3 for the batch sizes with the CDR and GDA datasets respectively, 100 for the dimension of the node/edge representation vectors in the inference layers). The values for the specific hyper-parameters of EoGANE (i.e., tuned with the development data for each dataset) include: $\gamma = 0.4$ for the controlling constant in the node representation computation of the inference layers (for both CDR and GDA), $\alpha_{rel} = 0.5$ and $\alpha_{const} = 0.1$ for the trade-off parameters in the overall loss function $\mathcal{L}$ for the CDR dataset, and $\alpha_{rel} = 0.4$ and $\alpha_{const} = 0.6$ for the GDA dataset.

Comparing to the State of the Art: This part compares the proposed model EoGANE with the state-of-the-art models for DRE. Table 1 reports the performance of the models on the CDR test set. In particular, the direct baseline for EoGANE is the EoG model in (Christopoulou et al., 2019) that also has the best-reported performance on the CDR dataset. In this table, we distinguish the models for DRE based on whether they use external knowledge/resources (i.e., the syntactic dependency tools) or not. As we can see from the table, among the models that do not rely on any external knowledge/resources (i.e., (Gu et al., 2017; Verga et al., 2018; Nguyen and Verspoor, 2018; Christopoulou et al., 2019) and the proposed model EoGANE), EoGANE achieves the best performance for all the overall, intra- and inter-sentence settings. In particular, EoGANE is 2.5% better than the second-best and direct baseline EoG on the absolute overall F1 score. This is significant with $p < 0.01$ and clearly demonstrates the effectiveness of the proposed model for DRE. Comparing to the models with external resources (e.g., additional training data, external tools) (Zhou et al., 2016; Zheng et al., 2018), EoGANE is also significantly better than the other models over different settings (i.e., the overall, inter-, and intra-sentence performance). The only exception is with the overall F1 score for the model in (Li et al., 2016b) that utilizes additional unlabeled training data. This further testifies to the benefits of EoGANE for DRE.

Model	**Overall**	**Intra**	**Inter**
(Gu et al., 2017)	61.3	57.2	11.7
(Verga et al., 2018)	62.1	-	-
(Nguyen and Verspoor, 2018)	62.3	-	-
EoG (Christopoulou et al., 2019)	63.6	68.2	50.9
EoGANE (our system)	**66.1**	**70.7**	**53.5**
(Zhou et al., 2016)*	61.3	-	-
(Peng et al., 2016)*	63.1	-	-
(Li et al., 2016b)*	67.7	58.9	-
(Chandrasekarasastry et al., 2018)*	60.3	65.1	45.7
(Zheng et al., 2018)*	61.5	-	-

Table 1: Overall, intra- and inter-sentence performance (i.e., F1 scores) of the models on the CDR test set. The methods with * employ additional training data or external tools.

Model	**Overall**	**Intra**	**Inter**
EoG (Christopoulou et al., 2019)	81.5	85.2	50.0
EoGANE (our system)	**82.8**	**86.3**	**58.6**

Table 2: Performance (i.e., F1 scores) on the GDA test set.

Finally, Table 2 shows the performance of EoG and EoGANE on the GDA test set. Note that in this dataset, EoG also has the best-reported performance. It is clear from the table that EoGANE is still significantly better than EoG for all performance settings (with $p < 0.01$), thus further confirming the advantage of EoGANE for DRE.

Ablation Study: There are three major components of EoGANE in this work, i.e., the node representation computation in the inference layers for $\mathcal{G}$ (called **NodeRep**), the $\mathcal{L}_{rel}$ loss term for the consistency of the node and edge representations in the inference layers, and the $\mathcal{L}_{const}$ loss term for the similarity of the mention representations. This part evaluates the contribution of these components by removing each of them from EoGANE and evaluating the performance of the remaining models. Note that if the node representation computation is not included in the inference layers (i.e., EoGANE-NodeRep), the two losses $\mathcal{L}_{rel}$ and $\mathcal{L}_{const}$ are also not used and the feature vectors V_{e_i,e_j} only involves the edge representation $\mathbf{e}^{2N}_{e_i,e_j}$ (i.e., $V_{e_i,e_j} = [\mathbf{e}^{2N}_{(e_i,e_j)}]$). In order to further shows the benefits of including the node representations in the inference layers, we also evaluate the performance of the "EoGANE-NodeRep" model when the initial embedding vectors for the nodes (i.e., $\mathbf{n}^1_i$) are incorporated into the feature vectors V_{e_i,e_j} for

Model	P	R	F1
EoGANE (full model)	63.4	64.3	**63.8**
EoGANE-$\mathcal{L}_{rel}$	60.6	61.7	61.1
EoGANE-$\mathcal{L}_{const}$	60.8	62.3	61.5
EoGANE-$\mathcal{L}_{const}$-$\mathcal{L}_{rel}$	61.1	60.6	60.8
EoGANE-NodeRep	58.6	60.8	59.7
EoGANE-NodeRep+Init	63.8	57.0	60.2

Table 3: The performance on the CDR development set.

prediction (i.e., $V_{e_i,e_j} = [\mathbf{n}^1_{e_i}, \mathbf{n}^1_{e_j}, \mathbf{e}^{2^N}_{(e_i,e_j)}]$) (called EoGANE-NodeRep+Init). Table 3 presents the overall performance of the models on the CDR development set. As we can see from the table, the elimination of any component in EoGANE would significantly hurt the performance, clearly verifying the effectiveness of the node representations and the proposed consistency constraints for DRE.

Analysis: In order to better understand the contribution of the node representation vectors in the proposed model EoGANE, we examine the examples in the CDR test set that are correctly predicted by EoGANE, but lead to incorrect predictions for EoG. Our analysis reveals that these examples tend to involve entities where the local/specific information of the individual entities is crucial to determine the relations between them. As EoGANE explicitly induces the node representations for the entities and include them in the feature vector for relation prediction, it can learn to capture those specific information of the entities to make correct predictions for these examples. Consider the following document (with two sentences) as an example:

"*The annual incidence of* ***warfarin***^{1}*-related bleeding at Brigham and Women's Hospital increased from 0.97/1,000 patient admissions in the first time period (January 1995 to October 1998) to 1.19/1,000 patient admissions in the second time period (November 1998 to August 2002) of this study. The proportion of patients with major and* ***intracranial bleeding***^{2} *increased from 20.2% and 1.9%, respectively, in the first time period , to 33.3% and 7.8%, respectively, in the second.*"

The two entities of interest in this document are "warfarin" (a chemical) and "intracranial bleeding" (a disease) whose entity mentions are in bold (i.e., "*warfarin*" and "*intracranial bleeding*"). In order to successfully predict the interaction relation between these two entities, the most important information for the models is that both entities are connected to the topic phrase "*warfarin-related bleeding*" of the document. In particular, for the entity "warfarin", the appearance of its only mention "*warfarin*" in the topic phrase "*warfarin-related bleeding*" in the document directly helps to identify "warfarin" as the chemical causing or being related to the bleeding in the phrase. Afterward, for the entity "intracranial bleeding", it also has only one mention in the document (i.e., "*intracranial bleeding*"). Based on the appearance of the word "*bleeding*" in both the only mention "*intracranial bleeding*" and the topic phrase "*warfarin-related bleeding*", we can infer that the entity "intracranial bleeding" is referring to the bleeding type expressed by the topic phrase in the document. Consequently, combining these evidences, we can conclude that the entity "intracranial bleeding" is caused by the entity "warfarin" in this case.

A notable point in our argument is that for both entities "warfarin" and "intracranial bleeding", their connections to the topic phrase can only be induced from the local context of their own entity mentions (i.e., the phrases "*warfarin-related bleeding*" and "*intracranial bleeding*"), highlighting the importance of the local/specific context of entity mention nodes for DRE. As EoGANE explicitly computes representation vectors for nodes in document graphs, it can learn to encode such local/specific context information of the entity mentions/entities into its representation vectors for the entities. These entity representation vectors, once incorporated into the feature vector V_{e_i,e_j} for relation prediction, can help the model to emphasize on these entity-specific information to do appropriate reasoning and produce correct prediction. This is in contrast to EoG that only focuses on the representation vectors of the edges, potentially blurring the information specific to the individual entities/entity mentions and failing to predict the relation in this case.

5 Conclusion

We present a novel deep learning model for DRE that explicitly computes the node representations for the document graphs in the graph-based edge-oriented models for DRE. This enables the models to better capture the specific information of the nodes and facilitates the incorporation of two novel consistency constraints to improve the representation vectors. The experiments demonstrate the effectiveness of the proposed method for DRE. In the future, we plan to apply the models in this work to the related tasks in information extraction.

References

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*.

Nagesh Panyam Chandrasekarasastry, Karin M. Verspoor, Trevor Cohn, and Kotagiri Ramamohanarao. 2018. Exploiting graph kernels for high performance biomedical relation extraction. In *Journal of Biomedical Semantics*, volume 9.

Billy Chiu, Gamal Crichton, Anna Korhonen, and Sampo Pyysalo. 2016. How to train good word embeddings for biomedical NLP. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. In *EMNLP-IJCNLP*.

Jinghang Gu, Fuqing Sun, Longhua Qian, and Guodong Zhou. 2017. Chemical-induced disease relation extraction via convolutional neural network. In *Database*.

Pankaj Gupta, Subburam Rajaram, Hinrich Schutze, and Thomas Runkler. 2019. Neural relation extraction within and across sentence boundaries. In *AAAI*.

Robin Jia, Cliff Wong, and Hoifung Poon. 2019. Document-level n-ary relation extraction with multiscale representation learning. In *NAACL-HLT*.

Xiaotian Jiang, Quan Wang, Peng Li, and Bin Wang. 2016. Relation extraction with multi-instance multi-label convolutional neural networks. In *COLING*.

Haodi Li, Ming Yang, Qingcai Chen, Buzhou Tang, Xiaolong Wang, and Jun Yan. 2018. Chemical-induced disease extraction via recurrent piecewise convolutional neural networks. In *BMC Medical Informatics and Decision Making*.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016a. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. In *Database*.

Zhiheng Li, Zhihao Yang, Hongfei Lin, Jian Wang, Yingyi Gui, Yin Zhang, and Lei Wang. 2016b. Cidextractor: A chemical-induced disease relation extraction system for biomedical literature. In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *ACL*.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *ACL*.

Dat Quoc Nguyen and Karin Verspoor. 2018. Convolutional neural networks for chemical-disease relation extraction are improved with character-based word embeddings. In *Proceedings of the BioNLP 2018 workshop*.

Thien Huu Nguyen and Ralph Grishman. 2014. Employing word representations and regularization for domain adaptation of relation extraction. In *ACL*.

Thien Huu Nguyen and Ralph Grishman. 2015. Relation extraction: Perspective from convolutional neural networks. In *The NAACL Workshop on Vector Space Modeling for NLP (VSM)*.

Thien Huu Nguyen and Ralph Grishman. 2016. Combining neural networks and log-linear models to improve relation extraction. In *Proceedings of IJCAI Workshop on Deep Learning for Artificial Intelligence (DLAI)*.

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

Yifan Peng, Chih-Hsuan Wei, and Zhiyong Lu. 2016. Improving chemical disease relation extraction with rich features and weakly labeled data. In *Journal of Cheminformatics*.

Chris Quirk and Hoifung Poon. 2017. Distant supervision for relation extraction beyond the sentence boundary. In *EACL*.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *ECML/PKDD*.

Sunil Kumar Sahu, Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Inter-sentence relation extraction with document-level graph convolutional neural network. In *ACL*.

Gaurav Singh and Parminder Bhatia. 2019. Relation extraction using explicit context conditioning. In *NAACL-HLT*.

Linfeng Song, Yue Zhang, Zhiguo Wang, and Daniel Gildea. 2018. N-ary relation extraction using graph-state LSTM. In *EMNLP*.

Shikhar Vashishth, Rishabh Joshi, Sai Suman Prayaga, Chiranjib Bhattacharyya, and Partha Talukdar. 2018. RESIDE: Improving distantly-supervised neural relation extraction using side information. In *EMNLP*.

Patrick Verga, Emma Strubell, and Andrew McCallum. 2018. Simultaneously self-attending to all mentions for full-abstract biological relation extraction. In *NAACL-HLT*.

Amir Pouran Ben Veyseh, Franck Dernoncourt, Dejing Dou, and Thien Huu Nguyen. 2020. Exploiting the syntax-model consistency for neural relation extraction. In *ACL*.

Amir Pouran Ben Veyseh, Thien Huu Nguyen, and Dejing Dou. 2019. Improving cross-domain performance for relation extraction via dependency prediction and information flow control. In *IJCAI*.

Wu Ye, Luo R., Leung H.C.M., Ting HF., and Lam TW. 2019. Renet: A deep learning approach for extracting gene-disease associations from literature. In *International Conference on Research in Computational Molecular Biology, pages 272–284. Springer*.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of machine learning research*, 3:1083–1106.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *EMNLP*.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *COLING*.

Wenyuan Zeng, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2017. Incorporating relation paths in neural relation extraction. In *EMNLP*.

Yuhao Zhang, Peng Qi, and Christopher D Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *EMNLP*.

Wei Zheng, Hongfei Lin, Zhiheng Li, Xiaoxia Liu, Zhengguang Li, Bo Xu, Yijia Zhang, Zhihao Yang, and Jian Wang. 2018. An effective neural model extracting document level chemical-induced disease relations from biomedical literature. In *Journal of biomedical informatics*.

Guodong Zhou, Jian Su, Jie Zhang, and Min Zhang. 2005. Exploring various knowledge in relation extraction. In *ACL*.

Huiwei Zhou, Huijie Deng, Long Chen, Yunlong Yang, Chen Jia, and Degen Huang. 2016. Exploiting syntactic and semantics information for chemical-disease relation extraction. In *Database: The Journal of Biological Databases and Curation*.