RESIDE: Improving Distantly-Supervised Neural Relation Extraction using Side Information

Shikhar Vashishth[1] **Rishabh Joshi**[2] * **Sai Suman Prayaga**[1]
Chiranjib Bhattacharyya[1] **Partha Talukdar**[1]
[1] Indian Institute of Science
[2] Birla Institute of Technology and Science, Pilani
{shikhar,chiru,ppt}@iisc.ac.in
f2014102@pilani.bits-pilani.ac.in, suman.sai14@gmail.com

Abstract

Distantly-supervised Relation Extraction (RE) methods train an extractor by automatically aligning relation instances in a Knowledge Base (KB) with unstructured text. In addition to relation instances, KBs often contain other relevant side information, such as aliases of relations (e.g., *founded* and *co-founded* are aliases for the relation *founderOfCompany*). RE models usually ignore such readily available side information. In this paper, we propose RESIDE, a distantly-supervised neural relation extraction method which utilizes additional side information from KBs for improved relation extraction. It uses entity type and relation alias information for imposing soft constraints while predicting relations. RESIDE employs Graph Convolution Networks (GCN) to encode syntactic information from text and improves performance even when limited side information is available. Through extensive experiments on benchmark datasets, we demonstrate RESIDE's effectiveness. We have made RESIDE's source code available to encourage reproducible research.

1 Introduction

The construction of large-scale Knowledge Bases (KBs) like Freebase (Bollacker et al., 2008) and Wikidata (Vrandečić and Krötzsch, 2014) has proven to be useful in many natural language processing (NLP) tasks like question-answering, web search, etc. However, these KBs are not exhaustive. Relation Extraction (RE) attempts to fill this gap by extracting semantic relationships between entity pairs from plain text. This task can be modeled as a simple classification problem after the entity pairs are specified. Formally, given an entity pair (e_1,e_2) from the KB and an entity annotated sentence (or instance), we aim to predict the relation r, from a predefined relation set, that exists between e_1 and e_2. If no relation exists, we simply label it *NA*.

Most supervised relation extraction methods require large labeled training data which is expensive to construct. Distant Supervision (DS) (Mintz et al., 2009) helps with the construction of this dataset automatically, under the assumption that if two entities have a relationship in a KB, then all sentences mentioning those entities express the same relation. While this approach works well in generating large amounts of training instances, the DS assumption does not hold in all cases. Riedel et al. (2010); Hoffmann et al. (2011); Surdeanu et al. (2012) propose multi-instance based learning to relax this assumption. However, they use NLP tools to extract features, which can be noisy.

Recently, neural models have demonstrated promising performance on RE. Zeng et al. (2014, 2015) employ Convolutional Neural Networks (CNN) to learn representations of instances. For alleviating noise in distant supervised datasets, attention has been utilized by (Lin et al., 2016; Jat et al., 2018). Syntactic information from dependency parses has been used by (Mintz et al., 2009; He et al., 2018) for capturing long-range dependencies between tokens. Recently proposed Graph Convolution Networks (GCN) (Defferrard et al., 2016) have been effectively employed for encoding this information (Marcheggiani and Titov, 2017; Bastings et al., 2017). However, all the above models rely only on the noisy instances from distant supervision for RE.

Relevant side information can be effective for improving RE. For instance, in the sentence, *Microsoft was started by Bill Gates.*, the type information of *Bill Gates (person)* and *Microsoft (organization)* can be helpful in predicting the correct relation *founderOfCompany*. This is because every relation constrains the type of its target en-

*Research internship at Indian Institute of Science.

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1257–1266
Brussels, Belgium, October 31 - November 4, 2018.

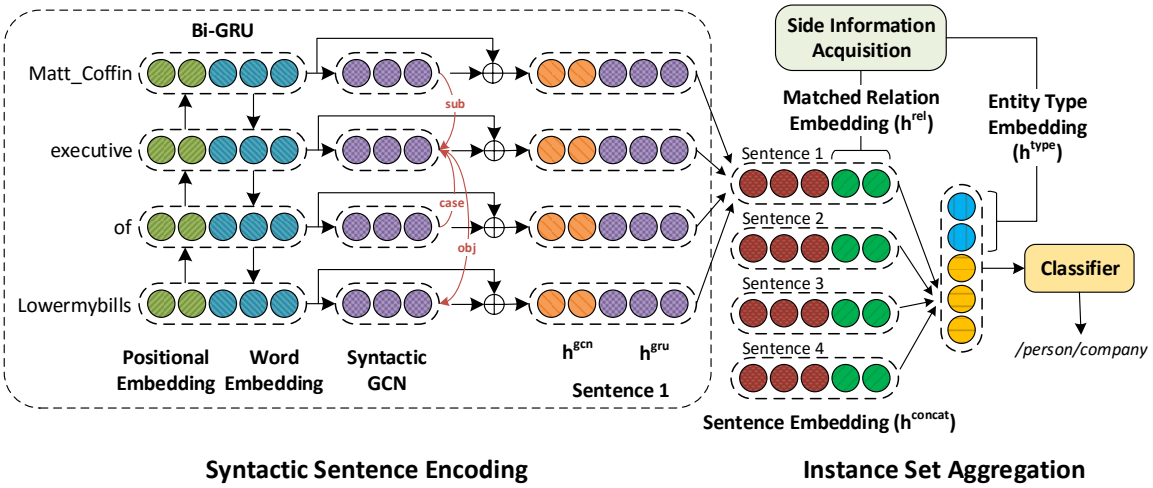


Figure 1: Overview of RESIDE. RESIDE first encodes each sentence in the bag by concatenating embeddings (denoted by $\oplus$) from Bi-GRU and Syntactic GCN for each token, followed by word attention. Then, sentence embedding is concatenated with relation alias information, which comes from the Side Information Acquisition Section (Figure 2), before computing attention over sentences. Finally, bag representation with entity type information is fed to a softmax classifier. Please see Section 5 for more details.

tities. Similarly, relation phrase *"was started by"* extracted using Open Information Extraction (Open IE) methods can be useful, given that the aliases of relation *founderOfCompany*, e.g., *founded, co-founded, etc.*, are available. KBs used for DS readily provide such information which has not been completely exploited by current models.

In this paper, we propose RESIDE, a novel distant supervised relation extraction method which utilizes additional supervision from KB through its neural network based architecture. RESIDE makes principled use of entity type and relation alias information from KBs, to impose soft constraints while predicting the relation. It uses encoded syntactic information obtained from Graph Convolution Networks (GCN), along with embedded side information, to improve neural relation extraction. Our contributions can be summarized as follows:

- We propose RESIDE, a novel neural method which utilizes additional supervision from KB in a principled manner for improving distant supervised RE.
- RESIDE uses Graph Convolution Networks (GCN) for modeling syntactic information and has been shown to perform competitively even with limited side information.
- Through extensive experiments on benchmark datasets, we demonstrate RESIDE's effectiveness over state-of-the-art baselines.

RESIDE's source code and datasets used in the paper are available at `http://github.com/malllabiisc/RESIDE`.

2 Related Work

Distant supervision: Relation extraction is the task of identifying the relationship between two entity mentions in a sentence. In supervised paradigm, the task is considered as a multi-class classification problem but suffers from lack of large labeled training data. To address this limitation, (Mintz et al., 2009) propose distant supervision (DS) assumption for creating large datasets, by heuristically aligning text to a given Knowledge Base (KB). As this assumption does not always hold true, some of the sentences might be wrongly labeled. To alleviate this shortcoming, Riedel et al. (2010) relax distant supervision for multi-instance single-label learning. Subsequently, for handling overlapping relations between entities (Hoffmann et al., 2011; Surdeanu et al., 2012) propose multi-instance multi-label learning paradigm.

Neural Relation Extraction: The performance of the above methods strongly rely on the quality of hand engineered features. Zeng et al. (2014)

propose an end-to-end CNN based method which could automatically capture relevant lexical and sentence level features. This method is further improved through piecewise max-pooling by (Zeng et al., 2015). Lin et al. (2016); Nagarajan et al. (2017) use attention (Bahdanau et al., 2014) for learning from multiple valid sentences. We also make use of attention for learning sentence and bag representations.

Dependency tree based features have been found to be relevant for relation extraction (Mintz et al., 2009). He et al. (2018) use them for getting promising results through a recursive tree-GRU based model. In RESIDE, we make use of recently proposed Graph Convolution Networks (Defferrard et al., 2016; Kipf and Welling, 2017), which have been found to be quite effective for modelling syntactic information (Marcheggiani and Titov, 2017; Nguyen and Grishman, 2018; Vashishth et al., 2018a).

Side Information in RE: Entity description from KB has been utilized for RE (Ji et al., 2017), but such information is not available for all entities. Type information of entities has been used by Ling and Weld (2012); Liu et al. (2014) as features in their model. Yaghoobzadeh et al. (2017) also attempt to mitigate noise in DS through their joint entity typing and relation extraction model. However, KBs like Freebase readily provide reliable type information which could be directly utilized. In our work, we make principled use of entity type and relation alias information obtained from KB. We also use unsupervised Open Information Extraction (Open IE) methods (Mausam et al., 2012; Angeli et al., 2015), which automatically discover possible relations without the need of any predefined ontology, which is used as a side information as defined in Section 5.2.

3 Background: Graph Convolution Networks (GCN)

In this section, we provide a brief overview of Graph Convolution Networks (GCN) for graphs with directed and labeled edges, as used in (Marcheggiani and Titov, 2017).

3.1 GCN on Labeled Directed Graph

For a directed graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ represent the set of vertices and edges respectively, an edge from node u to node v with label l_{uv} is represented as (u, v, l_{uv}). Since, information in directed edge does not necessarily propagate along its direction, following (Marcheggiani and Titov, 2017) we define an updated edge set $\mathcal{E}'$ which includes inverse edges (v, u, l_{uv}^{-1}) and self-loops $(u, u, \top)$ along with the original edge set $\mathcal{E}$, where $\top$ is a special symbol to denote self-loops. For each node v in $\mathcal{G}$, we have an initial representation $x_v \in \mathbb{R}^d, \forall v \in \mathcal{V}$. On employing GCN, we get an updated d-dimensional hidden representation $h_v \in \mathbb{R}^d, \forall v \in \mathcal{V}$, by considering only its immediate neighbors (Kipf and Welling, 2017). This can be formulated as:

$$h_v = f\left(\sum_{u \in \mathcal{N}(v)} \left(W_{l_{uv}} x_u + b_{l_{uv}}\right)\right).$$

Here, $W_{l_{uv}} \in \mathbb{R}^{d \times d}$ and $b_{l_{uv}} \in \mathbb{R}^d$ are label dependent model parameters which are trained based on the downstream task. $\mathcal{N}(v)$ refers to the set of neighbors of v based on $\mathcal{E}'$ and f is any non-linear activation function. In order to capture multi-hop neighborhood, multiple GCN layers can be stacked. Hidden representation of node v in this case after k^{th} GCN layer is given as:

$$h_v^{k+1} = f\left(\sum_{u \in \mathcal{N}(v)} \left(W_{l_{uv}}^k h_u^k + b_{l_{uv}}^k\right)\right).$$

3.2 Integrating Edge Importance

In automatically constructed graphs, some edges might be erroneous and hence need to be discarded. Edgewise gating in GCN by (Bastings et al., 2017; Marcheggiani and Titov, 2017) allows us to alleviate this problem by subduing the noisy edges. This is achieved by assigning a relevance score to each edge in the graph. At k^{th} layer, the importance of an edge (u, v, l_{uv}) is computed as:

$$g_{uv}^k = \sigma\left(h_u^k \cdot \hat{w}_{l_{uv}}^k + \hat{b}_{l_{uv}}^k\right), \tag{1}$$

Here, $\hat{w}_{l_{uv}}^k \in \mathbb{R}^m$ and $\hat{b}_{l_{uv}}^k \in \mathbb{R}$ are parameters which are trained and $\sigma(\cdot)$ is the sigmoid function. With edgewise gating, the final GCN embedding for a node v after k^{th} layer is given as:

$$h_v^{k+1} = f\left(\sum_{u \in \mathcal{N}(v)} g_{uv}^k \times \left(W_{l_{uv}}^k h_u^k + b_{l_{uv}}^k\right)\right). \tag{2}$$

4 RESIDE Overview

In multi-instance learning paradigm, we are given a bag of sentences (or instances) $\{s_1, s_2, ...s_n\}$ for a given entity pair, the task is to predict the relation between them. RESIDE consists of three components for learning a representation of a given bag, which is fed to a softmax classifier. We briefly present the components of RESIDE below. Each component will be described in detail in the subsequent sections. The overall architecture of RESIDE is shown in Figure 1.

1. **Syntactic Sentence Encoding:** RESIDE uses a Bi-GRU over the concatenated positional and word embedding for encoding the local context of each token. For capturing long-range dependencies, GCN over dependency tree is employed and its encoding is appended to the representation of each token. Finally, attention over tokens is used to subdue irrelevant tokens and get an embedding for the entire sentence. More details in Section 5.1.
2. **Side Information Acquisition:** In this module, we use additional supervision from KBs and utilize Open IE methods for getting relevant side information. This information is later utilized by the model as described in Section 5.2.
3. **Instance Set Aggregation:** In this part, sentence representation from syntactic sentence encoder is concatenated with the *matched relation embedding* obtained from the previous step. Then, using attention over sentences, a representation for the entire bag is learned. This is then concatenated with *entity type embedding* before feeding into the softmax classifier for relation prediction. Please refer to Section 5.3 for more details.

5 RESIDE Details

In this section, we provide the detailed description of the components of RESIDE.

5.1 Syntactic Sentence Encoding

For each sentence in the bag s_i with m tokens $\{w_1, w_2, ...w_m\}$, we first represent each token by k-dimensional GloVe embedding (Pennington et al., 2014). For incorporating relative position of tokens with respect to target entities, we use p-dimensional position embeddings, as used by (Zeng et al., 2014). The combined token embeddings are stacked together to get the sentence representation $\mathcal{H} \in \mathbb{R}^{m \times (k+2p)}$. Then, using Bi-GRU (Cho et al., 2014) over $\mathcal{H}$, we get the new sentence representation $\mathcal{H}^{gru} \in \mathbb{R}^{m \times d_{gru}}$, where d_{gru} is the hidden state dimension. Bi-GRUs have been found to be quite effective in encoding the context of tokens in several tasks (Sutskever et al., 2014; Graves et al., 2013).

Although Bi-GRU is capable of capturing local context, it fails to capture long-range dependencies which can be captured through dependency edges. Prior works (Mintz et al., 2009; He et al., 2018) have exploited features from syntactic dependency trees for improving relation extraction. Motivated by their work, we employ Syntactic Graph Convolution Networks for encoding this information. For a given sentence, we generate its dependency tree using Stanford CoreNLP (Manning et al., 2014). We then run GCN over the dependency graph and use Equation 2 for updating the embeddings, taking $\mathcal{H}^{gru}$ as the input. Since dependency graph has 55 different edge labels, incorporating all of them overparameterizes the model significantly. Therefore, following (Marcheggiani and Titov, 2017; Nguyen and Grishman, 2018; Vashishth et al., 2018a) we use only three edge labels based on the direction of the edge $\{$*forward* $(\rightarrow)$, *backward* $(\leftarrow)$, *self-loop* $(\top)\}$. We define the new edge label L_{uv} for an edge (u, v, l_{uv}) as follows:

$$L_{uv} = \begin{cases} \rightarrow & \text{if edge exists in dependency parse} \\ \leftarrow & \text{if edge is an inverse edge} \\ \top & \text{if edge is a self-loop} \end{cases}$$

For each token w_i, GCN embedding $h^{gcn}_{i_{k+1}} \in \mathbb{R}^{d_{gcn}}$ after k^{th} layer is defined as:

$$h^{gcn}_{i_{k+1}} = f\left(\sum_{u \in \mathcal{N}(i)} g^k_{iu} \times \left(W^k_{L_{iu}} h^{gcn}_{u_k} + b^k_{L_{iu}} \right) \right).$$

Here, g^k_{iu} denotes edgewise gating as defined in Equation 1 and L_{iu} refers to the edge label defined above. We use ReLU as activation function f, throughout our experiments. The syntactic graph encoding from GCN is appended to Bi-GRU output to get the final token representation, h^{concat}_i as $[h^{gru}_i; h^{gcn}_{i^{k+1}}]$. Since, not all tokens are equally relevant for RE task, we calculate the degree of relevance of each token using attention as used in

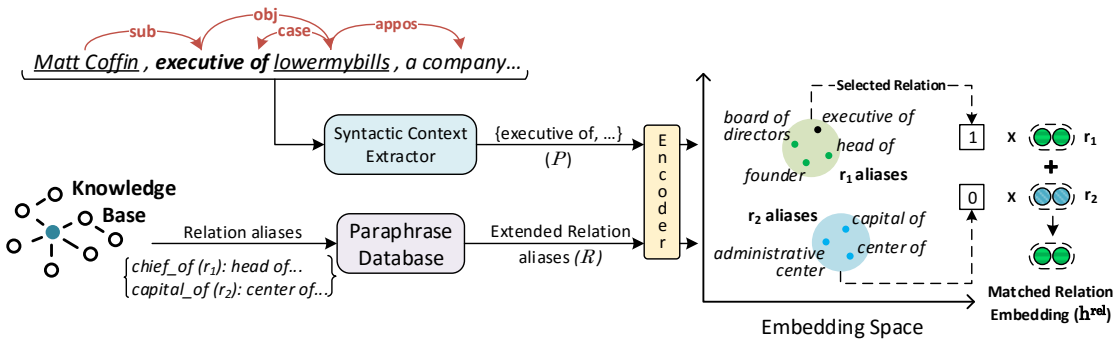


Figure 2: Relation alias side information extraction for a given sentence. First, Syntactic Context Extractor identifies relevant relation phrases $\mathcal{P}$ between target entities. They are then matched in the embedding space with the extended set of relation aliases $\mathcal{R}$ from KB. Finally, the relation embedding corresponding to the closest alias is taken as relation alias information. Please refer Section 5.2.

(Jat et al., 2018). For token w_i in the sentence, attention weight α_i is calculated as:

$$\alpha_i = \frac{\exp(u_i)}{\sum_{j=1}^{m} \exp(u_j)} \quad \text{where, } u_i = h_i^{concat} \cdot r.$$

where r is a random query vector and u_i is the relevance score assigned to each token. Attention values $\{\alpha_i\}$ are calculated by taking softmax over $\{u_i\}$. The representation of a sentence is given as a weighted sum of its tokens, $s = \sum_{j=1}^{m} \alpha_i h_i^{concat}$.

5.2 Side Information Acquisition

Relevant side information has been found to improve performance on several tasks (Ling and Weld, 2012; Vashishth et al., 2018b). In distant supervision based relation extraction, since the entities are from a KB, knowledge about them can be utilized to improve relation extraction. Moreover, several unsupervised relation extraction methods (Open IE) (Angeli et al., 2015; Mausam et al., 2012) allow extracting relation phrases between target entities without any predefined ontology and thus can be used to obtain relevant side information. In RESIDE, we employ Open IE methods and additional supervision from KB for improving neural relation extraction.

Relation Alias Side Information

RESIDE uses Stanford Open IE (Angeli et al., 2015) for extracting relation phrases between target entities, which we denote by $\mathcal{P}$. As shown in Figure 2, for the sentence *Matt Coffin, executive of lowermybills, a company..*, Open IE methods extract *"executive of"* between *Matt Coffin* and *lowermybills*. Further, we extend $\mathcal{P}$ by including tokens at one hop distance in dependency path from target entities. Such features from dependency parse have been exploited in the past by (Mintz et al., 2009; He et al., 2018). The degree of match between the extracted phrases in $\mathcal{P}$ and aliases of a relation can give important clues about the relevance of that relation for the sentence. Several KBs like Wikidata provide such relation aliases, which can be readily exploited. In RESIDE, we further expand the relation alias set using Paraphrase database (PPDB) (Pavlick et al., 2015). We note that even for cases when aliases for relations are not available, providing only the names of relations give competitive performance. We shall explore this point further in Section 7.3.

For matching $\mathcal{P}$ with the PPDB expanded relation alias set $\mathcal{R}$, we project both in a d-dimensional space using GloVe embeddings (Pennington et al., 2014). Projecting phrases using word embeddings helps to further expand these sets, as semantically similar words are closer in embedding space (Mikolov et al., 2013; Pennington et al., 2014). Then, for each phrase $p \in \mathcal{P}$, we calculate its cosine distance from all relation aliases in $\mathcal{R}$ and take the relation corresponding to the closest relation alias as a matched relation for the sentence. We use a threshold on cosine distance to remove noisy aliases. In RESIDE, we define a k_r-dimensional embedding for each relation which we call as *matched relation embedding* (h^{rel}). For a given sentence, h^{rel} is concatenated with its representa-

tion s, obtained from syntactic sentence encoder (Section 5.1) as shown in Figure 1. For sentences with $|\mathcal{P}| > 1$, we might get multiple matched relations. In such cases, we take the average of their embeddings. We hypothesize that this helps in improving the performance and find it to be true as shown in Section 7.

Entity Type Side Information

Type information of target entities has been shown to give promising results on relation extraction (Ling and Weld, 2012; Yaghoobzadeh et al., 2017). Every relation puts some constraint on the type of entities which can be its subject and object. For example, the relation *person/place_of_birth* can only occur between a *person* and a *location*. Sentences in distance supervision are based on entities in KBs, where the type information is readily available.

In RESIDE, we use types defined by FIGER (Ling and Weld, 2012) for entities in Freebase. For each type, we define a k_t-dimensional embedding which we call as *entity type embedding* (h^{type}). For cases when an entity has multiple types in different contexts, for instance, *Paris* may have types *government* and *location*, we take the average over the embeddings of each type. We concatenate the *entity type embedding* of target entities to the final bag representation before using it for relation classification. To avoid over-parameterization, instead of using all fine-grained 112 entity types, we use 38 coarse types which form the first hierarchy of FIGER types.

5.3 Instance Set Aggregation

For utilizing all valid sentences, following (Lin et al., 2016; Jat et al., 2018), we use attention over sentences to obtain a representation for the entire bag. Instead of directly using the sentence representation s_i from Section 5.1, we concatenate the embedding of each sentence with *matched relation embedding* h_i^{rel} as obtained from Section 5.2. The attention score α_i for i^{th} sentence is formulated as:

$$\alpha_i = \frac{\exp(\hat{s}_i \cdot q)}{\sum_{j=1}^{n} \exp(\hat{s}_j \cdot q)} \quad \text{where, } \hat{s}_i = [s_i; h_i^{rel}].$$

here q denotes a random query vector. The bag representation $\mathcal{B}$, which is the weighted sum of its sentences, is then concatenated with the *entity type embeddings* of the subject (h_{sub}^{type}) and object (h_{obj}^{type}) from Section 5.2 to obtain $\hat{\mathcal{B}}$.

$$\hat{\mathcal{B}} = [\mathcal{B}; h_{sub}^{type}; h_{obj}^{type}] \quad \text{where, } \mathcal{B} = \sum_{i=1}^{n} \alpha_i \hat{s}_i.$$

Finally, $\hat{\mathcal{B}}$ is fed to a softmax classifier to get the probability distribution over the relations.

$$p(y) = \text{Softmax}(W \cdot \hat{\mathcal{B}} + b).$$

Datasets	Split	# Sentences	# Entity-pairs
Riedel (# Relations: 53)	Train	455,771	233,064
	Valid	114,317	58,635
	Test	172,448	96,678
GIDS (# Relations: 5)	Train	11,297	6,498
	Valid	1,864	1,082
	Test	5,663	3,247

Table 1: Details of datasets used. Please see Section 6.1 for more details.

6 Experimental Setup

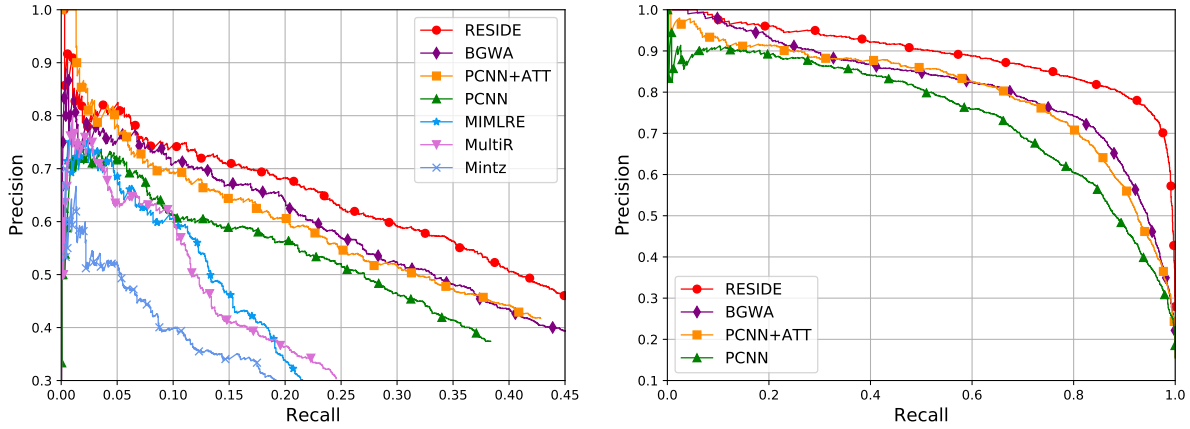
6.1 Datasets

In our experiments, we evaluate the models on Riedel and Google Distant Supervision (GIDS) dataset. Statistics of the datasets is summarized in Table 1. Below we described each in detail[1].

1. **Riedel:** The dataset is developed by (Riedel et al., 2010) by aligning Freebase relations with New York Times (NYT) corpus, where sentences from the year 2005-2006 are used for creating the training set and from the year 2007 for the test set. The entity mentions are annotated using Stanford NER (Finkel et al., 2005) and are linked to Freebase. The dataset has been widely used for RE by (Hoffmann et al., 2011; Surdeanu et al., 2012) and more recently by (Lin et al., 2016; Feng et al.; He et al., 2018).

2. **GIDS:** Jat et al. (2018) created Google Distant Supervision (GIDS) dataset by extending the Google relation extraction corpus[2] with additional instances for each entity pair. The dataset assures that the at-least-one assumption of multi-instance learning, holds. This makes automatic evaluation more reliable and thus removes the need for manual verification.

[1]Data splits and hyperparameters are in supplementary.
[2]https://research.googleblog.com/2013/04/50000-lessons-on-how-to-read-relation.html



(a) Riedel dataset

(b) GIDS dataset

Figure 3: Comparison of Precision-recall curve. RESIDE achieves higher precision over the entire range of recall than all the baselines on both datasets. Please refer Section 7.1 for more details.

6.2 Baselines

For evaluating RESIDE, we compare against the following baselines:

- **Mintz:** Multi-class logistic regression model proposed by (Mintz et al., 2009) for distant supervision paradigm.
- **MultiR:** Probabilistic graphical model for multi instance learning by (Hoffmann et al., 2011)
- **MIMLRE:** A graphical model which jointly models multiple instances and multiple labels. More details in (Surdeanu et al., 2012).
- **PCNN:** A CNN based relation extraction model by (Zeng et al., 2015) which uses piecewise max-pooling for sentence representation.
- **PCNN+ATT:** A piecewise max-pooling over CNN based model which is used by (Lin et al., 2016) to get sentence representation followed by attention over sentences.
- **BGWA:** Bi-GRU based relation extraction model with word and sentence level attention (Jat et al., 2018).
- **RESIDE:** The method proposed in this paper, please refer Section 5 for more details.

6.3 Evaluation Criteria

Following the prior works (Lin et al., 2016; Feng et al.), we evaluate the models using held-out evaluation scheme. This is done by comparing the relations discovered from test articles with those in Freebase. We evaluate the performance of models with Precision-Recall curve and top-N precision (P@N) metric in our experiments.

7 Results

In this section we attempt to answer the following questions:

Q1. Is RESIDE more effective than existing approaches for distant supervised RE? (7.1)

Q2. What is the effect of ablating different components on RESIDE's performance? (7.2)

Q3. How is the performance affected in the absence of relation alias information? (7.3)

7.1 Performance Comparison

For evaluating the effectiveness of our proposed method, RESIDE, we compare it against the baselines stated in Section 6.2. We use only the neural baselines on GIDS dataset. The Precision-Recall curves on Riedel and GIDS are presented in Figure 3. Overall, we find that RESIDE achieves higher precision over the entire recall range on both the datasets. All the non-neural baselines could not perform well as the features used by them are mostly derived from NLP tools which can be erroneous. RESIDE outperforms PCNN+ATT and BGWA which indicates that incorporating side information helps in improving the performance of the model. The higher performance of BGWA and PCNN+ATT over PCNN shows that attention helps in distant supervised RE. Following (Lin et al., 2016; Liu et al., 2017), we also evaluate our method with different number of sentences. Results summarized in Table 2, show the improved precision of RESIDE in all test settings, as compared to the neural baselines, which demonstrates

	One			Two			All		
	P@100	P@200	P@300	P@100	P@200	P@300	P@100	P@200	P@300
PCNN	73.3	64.8	56.8	70.3	67.2	63.1	72.3	69.7	64.1
PCNN+ATT	73.3	69.2	60.8	77.2	71.6	66.1	76.2	73.1	67.4
BGWA	78.0	71.0	63.3	81.0	73.0	64.0	82.0	75.0	72.0
RESIDE	**80.0**	**75.5**	**69.3**	**83.0**	**73.5**	**70.6**	**84.0**	**78.5**	**75.6**

Table 2: P@N for relation extraction using variable number of sentences in bags (with more than one sentence) in Riedel dataset. Here, One, Two and All represents the number of sentences randomly selected from a bag. RESIDE attains improved precision in all settings. More details in Section 7.1

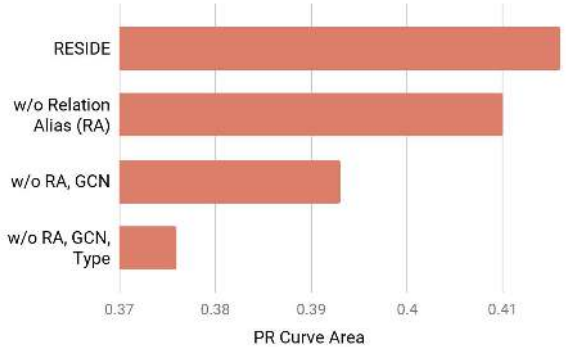


Figure 4: Performance comparison of different ablated version of RESIDE on Riedel dataset. Overall, GCN and side information helps RESIDE improve performance. Refer Section 7.2.

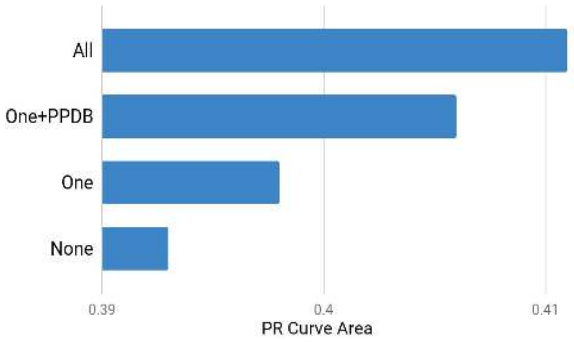


Figure 5: Performance on settings defined in Section 7.3 with respect to the presence of relation alias side information on Riedel dataset. RESIDE performs comparably in the absence of relations from KB.

the efficacy of our model.

7.2 Ablation Results

In this section, we analyze the effect of various components of RESIDE on its performance. For this, we evaluate various versions of our model with cumulatively removed components. The experimental results are presented in Figure 4. We observe that on removing different components from RESIDE, the performance of the model degrades drastically. The results validate that GCNs are effective at encoding syntactic information. Further, the improvement from side information shows that it is complementary to the features extracted from text, thus validating the central thesis of this paper, that inducing side information leads to improved relation extraction.

7.3 Effect of Relation Alias Side Information

In this section, we test the performance of the model in setting where relation alias information is not readily available. For this, we evaluate the performance of the model on four different settings:

- **None:** Relation aliases are not available.
- **One:** The name of relation is used as its alias.
- **One+PPDB:** Relation name extended using Paraphrase Database (PPDB).
- **All:** Relation aliases from Knowledge Base[3]

The overall results are summarized in Figure 5. We find that the model performs best when aliases are provided by the KB itself. Overall, we find that RESIDE gives competitive performance even when very limited amount of relation alias information is available. We observe that performance improves further with the availability of more alias information.

8 Conclusion

In this paper, we propose RESIDE, a novel neural network based model which makes principled use of relevant side information, such as entity type and relation alias, from Knowledge Base, for improving distant supervised relation extraction. RESIDE employs Graph Convolution Networks for

[3]Each relation in Riedel dataset is manually mapped to corresponding Wikidata property for getting relation aliases. Few examples are presented in supplementary material.

encoding syntactic information of sentences and is robust to limited side information. Through extensive experiments on benchmark datasets, we demonstrate RESIDE's effectiveness over state-of-the-art baselines. We have made RESIDE's source code publicly available to promote reproducible research.

Acknowledgements

We thank the anonymous reviewers for their constructive comments. This work is supported in part by the Ministry of Human Resource Development (Government of India), CAIR (DRDO) and by a gift from Google.

References

Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *ACL (1)*, pages 344–354. The Association for Computer Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv e-prints*, abs/1409.0473.

Joost Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, and Khalil Simaan. 2017. Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967, Copenhagen, Denmark. Association for Computational Linguistics.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 1247–1250, New York, NY, USA. ACM.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734. Association for Computational Linguistics.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 3844–3852, USA. Curran Associates Inc.

Xiaocheng Feng, Jiang Guo, Bing Qin, Ting Liu, and Yongjie Liu. Effective deep memory networks for distant supervised relation extraction.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics.

A. Graves, A. r. Mohamed, and G. Hinton. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649.

Zhengqiu He, Wenliang Chen, Zhenghua Li, Meishan Zhang, Wei Zhang, and Min Zhang. 2018. See: Syntax-aware entity embedding for neural relation extraction.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 541–550. Association for Computational Linguistics.

S. Jat, S. Khandelwal, and P. Talukdar. 2018. Improving Distantly Supervised Relation Extraction using Word and Entity Based Attention. *ArXiv e-prints*.

Guoliang Ji, Kang Liu, Shizhu He, and Jun Zhao. 2017. Distant supervision for relation extraction with sentence-level attention and entity descriptions. In *AAAI*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2124–2133.

Xiao Ling and Daniel S. Weld. 2012. Fine-grained entity recognition. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, pages 94–100. AAAI Press.

Tianyu Liu, Kexiang Wang, Baobao Chang, and Zhifang Sui. 2017. A soft-label method for noise-tolerant distantly supervised relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1790–1795. Association for Computational Linguistics.

Yang Liu, Kang Liu, Liheng Xu, and Jun Zhao. 2014. Exploring fine-grained entity type constraints for distantly supervised relation extraction. In *COLING*.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. *CoRR*, abs/1703.04826.

Mausam, Michael Schmitz, Robert Bart, Stephen Soderland, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 523–534, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 3111–3119, USA. Curran Associates Inc.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Tushar Nagarajan, Sharmistha Jat, and Partha Talukdar. 2017. Candis: Coupled & attention-driven neural distant supervision. *CoRR*, abs/1710.09942.

Thien Nguyen and Ralph Grishman. 2018. Graph convolutional networks with argument-aware pooling for event detection.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*, pages 425–430.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 455–465. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Shikhar Vashishth, Shib Sankar Dasgupta, Swayambhu Nath Ray, and Partha Talukdar. 2018a. Dating documents using graph convolution networks. In *56th Annual Meeting of the Association for Computational Linguistics (ACL 2018), July 15-20, 2018, Melbourne, Australia*.

Shikhar Vashishth, Prince Jain, and Partha Talukdar. 2018b. Cesi: Canonicalizing open knowledge bases using embeddings and side information. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pages 1317–1327, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Yadollah Yaghoobzadeh, Heike Adel, and Hinrich Schütze. 2017. Noise mitigation for neural entity typing and relation extraction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1183–1194. Association for Computational Linguistics.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2335–2344.