Zero-shot Relation Classification from Side Information

Jiaying Gong
Virginia Tech
Blacksburg, U.S.
gjiaying@vt.edu

Hoda Eldardiry
Virginia Tech
Blacksburg, U.S.
hdardiry@vt.edu

ABSTRACT

We propose a zero-shot learning relation classification (ZSLRC) framework that improves on state-of-the-art by its ability to recognize novel relations that were not present in training data. The zero-shot learning approach mimics the way humans learn and recognize new concepts with no prior knowledge. To achieve this, ZSLRC uses advanced prototypical networks that are modified to utilize weighted side (auxiliary) information. ZSLRC's side information is built from keywords, hypernyms of name entities, and labels and their synonyms. ZSLRC also includes an automatic hypernym extraction framework that acquires hypernyms of various name entities directly from the web. ZSLRC improves on state-of-the-art few-shot learning relation classification methods that rely on labeled training data and is therefore applicable more widely even in real-world scenarios where some relations have no corresponding labeled examples for training. We present results using extensive experiments on two public datasets (NYT and FewRel) and show that ZSLRC significantly outperforms state-of-the-art methods on supervised learning, few-shot learning, and zero-shot learning tasks. Our experimental results also demonstrate the effectiveness and robustness of our proposed model.

CCS CONCEPTS

• **Computing methodologies → Natural language processing**; Machine learning; *Information extraction.*

KEYWORDS

relation classification; zero-shot learning; side information acquisition; prototypical network

ACM Reference Format:
Jiaying Gong and Hoda Eldardiry. 2021. Zero-shot Relation Classification from Side Information. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3459637.3482403

1 INTRODUCTION

Relation classification aims to infer the relation between two name entities in a sentence. Supervised learning methods for relation classification have been widely used to classify relations based on training labeled data. Distant supervision or crowdsourcing have been used to collect more examples with labels and train the model for relation classification. However, these methods are limited by the quantity (for supervised) and quality (for distant-supervised) of the training data because manually labeling the data is time-consuming and labor-intensive, and data labeled by distant-supervision is noisy. To overcome the problem of insufficient high-quality data, few-shot learning has been designed to require only few labeled sentences for training. A lot of research has been done on few-shot learning for computer vision [18, 21, 39], and some work also includes few-shot learning methods for relation classification [7, 9, 13]. However, these works still require a few instances for training, and they still do not work when no training instances are available.

Some work on open information extraction (OpenIE) discovers new relationships in open-domain corpora without labeling the data [1]. OpenIE aims to extract relation phrases directly from the text. However, this technique can not effectively select meaningful relation patterns and discard irrelevant information. Besides, this technique can not discover relations if the relation's name does not appear in the given sentence. For example, OpenIE can not identify the relation of the sentence in Figure 1.

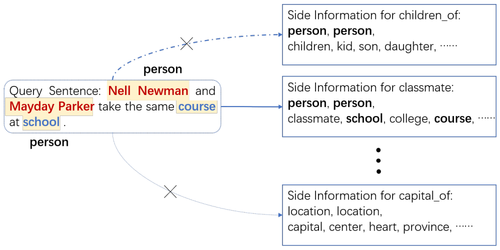

Side Information for children_of:
person, person,
children, kid, son, daughter, ······
person
Query Sentence: Nell Newman and Mayday Parker take the same course at school .
person
Side Information for classmate:
person, person,
classmate, school, college, course, ······
Side Information for capital_of:
location, location,
capital, center, heart, province, ······

Figure 1: Example of relation classification based on side information.

To address the limitations mentioned above, we focus on relation classification in the context of zero-shot learning. Zero-shot learning (ZSL) is similar to the way humans learn and recognize new concepts. It is a novel learning technique that does not use any exemplars of the unseen categories during training. We propose a zero-shot learning model for relation classification (ZSLRC), which focuses on recognizing new relations with no corresponding labeled data available for training. ZSLRC is modified on prototypical networks utilizing side (auxiliary) information. We construct weighted side information from labels and their synonyms, hypernyms of two name entities, and keywords from training sentences. The ZSL-based model can recognize new relations based on the side information available for it instead of using a collection of labeled sentences. We incorporate side information to enable our model

CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482403

to identify relations that never appear in the training datasets. We also build an automatic hypernym extraction framework to help us acquire hypernyms of different entities directly from the web. Details of side information construction are described in Section 3.2.

Figure 1 shows an example of how side information can be used for classifying relations. Different side information is given for different relations. The query sentence in the example has a relation of *classmate_of*, but the word classmate never appears in the sentence. We first get the two name entities *Nell Newman* and *Mayday Parker* of the sentence and extract the hypernyms of the name entities *person* and *person* based on our proposed hypernym extraction module in Section 3.2.1. In this example, relation *capital_of* is eliminated because the hypernyms of *capital_of* should be *location* and *location*. Then we extract the keywords *course* and *school* from the query sentence and compare the distance with the keywords in the side information box. In this way, relation *children_of* is eliminated.

To make relation classification effective in real-world scenarios, we design our model with the ability of classifying both relations with training instances and relations without any training instances. We modify the vanilla prototypical networks to deal with both scenarios and compare the distance between the query sentence and the weighted prototype. If the exponential of the minus distance is above a threshold, we consider the query sentence has a new relation. For new relations identification, we take the side information embedding from the query sentence and compare the distance of it with the side information embedding of new relations. We conduct different experiments on both a noisy and a clean dataset and adding different percentages of new relations to evaluate the effectiveness and robustness of our proposed model. Besides, we also evaluate our proposed model in supervised learning, few-shot learning, and zero-shot learning tasks. The results show that our proposed model outperforms other existing models in all three tasks. The contributions of this paper can be summarized as follows:

- We propose the first approach (ZSLRC) to enable zero-shot learning on relation classification without relying on other complex models that need to be learned and assumed to be 100% accurate.
- ZSLRC uses side information including labels, keywords, and hypernyms of name entities, and it has been shown that our model can perform competitively using the weighted side information. We build an automatic hypernym extraction framework to extract hypernyms of words from the web.
- We modify prototypical networks to recognize new relations in addition to recognized previously known relations. Results show the effectiveness and robustness of our modified prototypical networks in different learning tasks.
- We demonstrate that our proposed model significantly outperforms state-of-the-art methods on supervised learning, few-shot learning, and zero-shot learning tasks. We ran extensive experiments on two datasets.

2 RELATED WORK

Supervised Relation Classification. Relation Classification aims to classify relations between entities. Many existing relation classification methods are based on supervised learning, where neural networks are used to extract semantic features from text automatically. For example, convolutional neural networks (CNNs) are used to learn textual patterns [6, 23, 28, 36, 43, 49]. Recurrent neural networks (RNNs) are used to better capture the sequential information present in the input data [27, 44, 47]. Graph neural networks (GNNs) are used to find dependencies and capture long-range relations between words [46, 48]. Although these traditional Relation Classification methods have achieved promising results by taking advantage of supervised or distantly-supervised data, they exhibit a fundamental limitation since they all need large quantities of labeled training data.

Open Relation Extraction. Many existing approaches focus on discovering new relationships in open-domain corpora. This is because traditional supervised RC can not find new relation types due to their limited ability to classify predefined relation types. Open RE or Open information extraction (OpenIE) aims to extract relation phrases directly from the text. For example, tagging-based methods [3, 15] and clustering-based methods [25, 37] are used to discover new relation types. Other work proposed Relational Siamese Networks to transfer relational knowledge from supervised OpenRE data to calculate similarity of unlabeled sentences for open relation clustering [37]. However, OpenRE can not effectively select meaningful relation patterns and discard irrelevant information. In the real world, methods that rely on predefined relation types are always known to lack of training data.

Zero-shot Learning. Zero-shot learning has been widely applied in computer vision [2, 14, 16, 20, 31, 38, 41]. Similar to zero-shot learning, few shot learning is well-studied in the field of relation classification [5, 7–9, 40, 42]. However, compared with zero-shot learning for computer vision and few-shot learning explored in relation classification, there exists little work towards zero-shot learning in the domain of natural language processing. Some current work uses a transferable architecture to jointly represent and map event types in order to detect unseen event types [12]. Other work proposed a zero-shot learning method for relation extraction from webpages with unseen templates [24]. However, this method solves a different problem, only predicting relation types in unseen structures of webpages instead of new relation types. The most related work to zero-shot learning for relation classification uses zero-shot learning to extract unseen relation types by listing questions that define the relation's slot values [17]. However, this method requires external help, such as a question-answering dataset annotated by a human. In addition, this method assumes that (1) a good reading comprehension model is learned and that (2) all values extracted from this model are correct. In contrast, our proposed model can identify new relation types without training sentences and does not need to rely on other models. We construct weighted side information to train the model without labeled training sentences. For example, some previous works use side information from knowledge graph or label to lower the noise and improve performance in distantly-supervised relation classification [11, 35].

3 METHODOLOGY

In this section, we introduce the overview of ZSLRC model. Figure 2 shows the architecture of zero-shot learning for relation classification. It consists of three parts: Sentence Encoder, Side Information

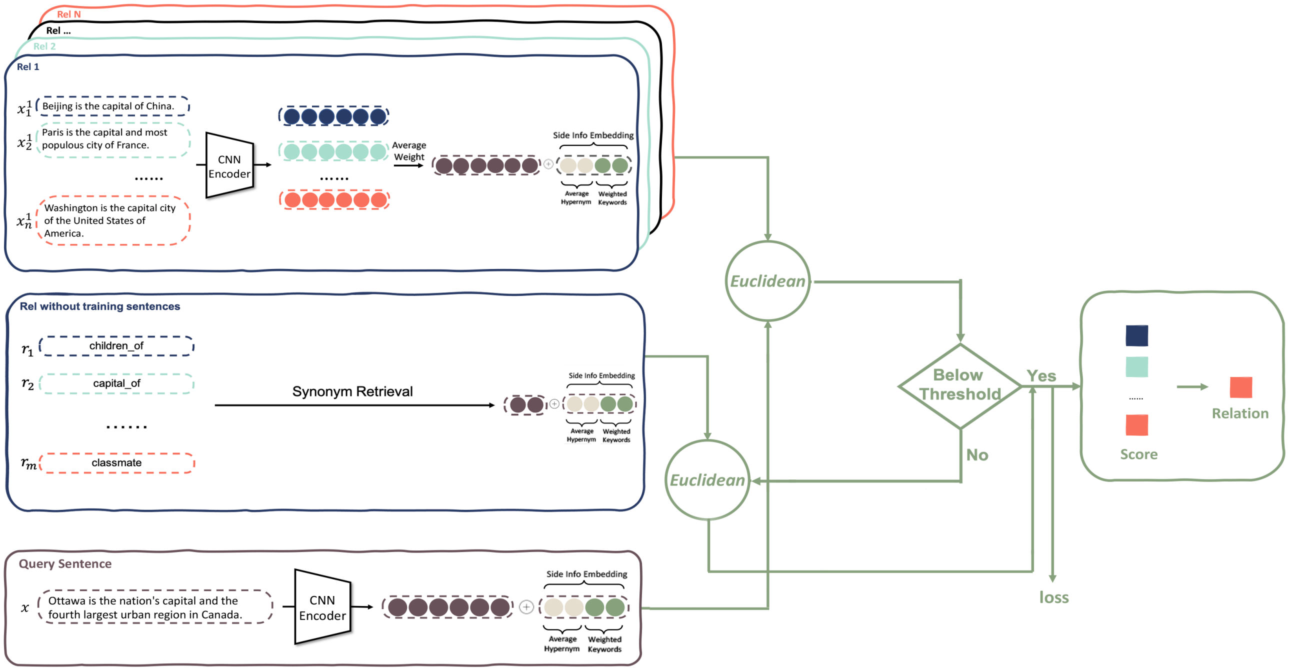


Figure 2: Model of Zero-shot Learning for Relation Classification (ZSLRC)

Extraction and Prototypical Network with Weighted Side Information Embedding. We describe these parts in detail below.

3.1 Sentence Encoder

The inputs of ZSLRC model are a set of sentences $\{x_1, x_2, x_3, \cdots x_n\}$ and its corresponding entity pair. For relations with training sentences, our model measures the probability of each relation r' by measuring the distance between query sentences and the average weight of training sentence embeddings. For relations without training sentences, the probability of r' is done by measuring the distance between side information from query sentences and side information from relation types.

3.1.1 ***Word Embeddings.*** Word embeddings aim to map words or phrases from vocabulary to vectors of numerical forms. The distributed representations are learned based on the usage of words, which allows words that are used in similar ways to result in having similar representations, naturally capturing syntactic and semantic meanings of the words. In this paper, we first tokenize and lemmatize all words in a sentence, and a 50-dimension GloVe, a pre-trained global log-bilinear regression model for the unsupervised learning of word representations, is used as our initial word embeddings [30]. If the words are out of vocabulary, they are randomly embedded first, and the vectors are updated while the model is training. Word embedding vectors are updated through training the model.

3.1.2 ***Position Embeddings.*** Word positions also play an essential role in relation classification. Words closer to name entities have more influence on the determination of relation types. We use position features, a combination of relative distances from current word to both entities, to identify entity pairs [43]. After concatenating position embeddings and word embeddings, the vector representation transforms a sentence into a matrix $S \in \mathbb{R}^{s \times d}$, where s is the sentence length and $d = d_w + d_p \times 2$. For each word $w \in S = \{w_1, w_2, \cdots w_n\}$, its embedding $\hat{w}_i$ is initialized as follows:

$$\hat{w}_i = w_i \oplus p_{i1} \oplus p_{i2} \tag{1}$$

where w_i is the pre-trained word vector and p_{i1}, p_{i2} are two corresponding position embeddings of the current word with two name entities. Symbol $\oplus$ indicates the concatenation operator. The matrix S is then fed into the CNN encoder.

3.1.3 ***CNN Encoder.*** Because convolutional neural networks can merge all local features and perform the prediction globally, we choose CNN to encode our input embeddings. We learn the instance embedding as follows:

$$x_i = CNN(w_{i-\frac{n-1}{2}}, \cdots, w_{i+\frac{n-1}{2}}) \tag{2}$$

$$\hat{x_i} = max(0, x_i) \tag{3}$$

$$[s]_j = max\left\{[\hat{x_1}]_j, \cdots, [\hat{x_n}]_j\right\} \tag{4}$$

where $CNN(\cdot)$ is a convolutional layer with window size n over the word sequence. A non-linear activation function ReLU is added after the convolutional layer. Function max denotes max-pooling and $[\cdot]_j$ is the j-th value of a vector.

Figure 3 shows the architecture of CNN encoder used in this paper. Due to time complexity, We simply use one convolutional layer, one non-linear layer, and one max pooling layer to get the sentence embedding. The parameter settings are described in Section 4.3

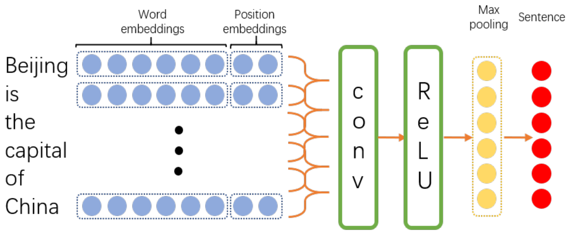


Figure 3: CNN Encoder

3.1.4 ***Side Information Embeddings.*** Label information and keywords in each sentence also play an essential role in improving the performance of relation classification. For relations without any training sentences, hypernyms, labels and their corresponding synonyms are used as side information. Side information embeddings are concatenated to the prototype for each relation after CNN encoder. The final prototype including side information for each relation can be expressed as follows:

$$c_i{}' = \begin{cases} r \oplus si_h \oplus si_r \oplus si_k & r \neq 0 \\ si_h \oplus si_r \oplus si_s & r = 0 \end{cases} \tag{5}$$

where r is the initial prototype for each relation, si_h represents the side information from hypernyms, si_r is the side information from relation types, si_k is the side information from keywords in all training sentences of one relation type and si_s is the synonyms for relation types. Details for side information description and its extraction will be described in Section 3.2.

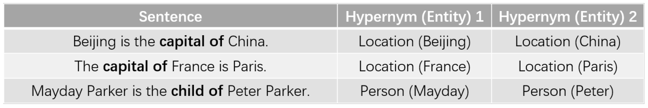
3.2 Side Information Extraction

Side information is the auxiliary information used to detect new relation types. For relations with training sentences, side information is the hypernyms of two entities, relationship between two entities, and keywords from all training sentences with the same relation type. For relations without training sentences, the side information is hypernyms of two entities by manually labeling, relation type itself, and synonyms of the relation types. For query sentences, the side information is hypernyms of two entities and keywords extracted from the sentence.

In this section, we describe hypernyms extraction and keyword extraction in detail because the relationship can be easily obtained from labels, and synonyms of relation types can also be easily acquired through WordNet or other dictionaries [26].

3.2.1 ***Hypernyms Extraction.*** A hypernym is the broad meaning of more specific words. For example, an animal is a hypernym of a dog. The hypernym of two entities in one sentence is extremely important for relation classification. Figure 4 shows an example of different sentences with different hypernyms, indicating that hypernyms can help classify different relation types. For example, relation *capital_of* can only occur between two locations, and relation *child_of* can only occur between two people.

Hypernyms of entities are not easy to acquire. Some existing tools, such as WordNet can only acquire hypernyms from limited vocabularies. In our experiments, less than 10% of entities can achieve their corresponding hypernyms through WordNet. Some previous works used entity types (hypernyms) defined by FIGER as side information [22, 35]. However, only 112 entity types are provided by FIGER, and only 38 of them are used as entity types by [35]. Most of the name entities from sentences in the real world can not get their hypernyms based on this list due to its fixed size and limited entity types. Therefore, we provide an approach for extracting hypernyms through external help from the web.



Sentence	Hypernym (Entity) 1	Hypernym (Entity) 2
Beijing is the **capital of** China.	Location (Beijing)	Location (China)
The **capital of** France is Paris.	Location (France)	Location (Paris)
Mayday Parker is the **child of** Peter Parker.	Person (Mayday)	Person (Peter)

Figure 4: Example of sentences with different hypernyms.

Hypernyms can be discovered through the definition of entities. We build an automatic hypernym extraction framework based on WordNet, Merriam Webster [1] and Wikidata [2]. Merriam Webster includes a part of speech description to distinguish nouns of a person (biographical) from nouns of location (geographical). In the real world, there are quite a number of relations that occur between these two hypernyms. Wikidata provides definitions for different entities. We crawl the definition for each entity through Wikidata and get the first Noun as hypernym. For example, *Jeff Bezos is the founder of Amazon.* Commerce is extracted as a hypernym for Amazon. Most entities, including person, location or other nouns, can get their hypernyms through our proposed framework. The entire framework of hypernym extraction is described in detail in Algorithm 1.

```
Algorithm 1: Hypernym Extraction
  Input   :sentences $\{x_1, x_2, x_3, \cdots x_n\}$ with same relation.
  Output:hypernyms of two entities from one relation.
  Step 1: Initialize hypernyms to none.
  Step 2: Find hypernyms $\{h_1^1, h_1^2, \cdots h_1^n\}$ and $\{h_2^1, h_2^2, \cdots h_2^n\}$
    of entities from WordNet.
  Step 3: $h_1 = major\ \{h_1^1, \cdots h_1^n\}$, $h_2 = major\ \{h_2^1, \cdots h_2^n\}$.
  if $h$ == none then
      go to Step 4.
  else
      End
  Step 4: Getting PoS descriptions $PD$ of entities $E =$
    $\{e_1^1, \cdots e_1^n\}$ and $\{e_2^1, \cdots e_2^n\}$ from Merriam Webster.
    $h = Tokenize(PD)$
  if $h$ == none then
      go to Step 5.
  else
      End
  Step 5: Crawling definitions $D$ for $E$ from Wikidata. $h$ = first
    Noun of $Tokenize(D)$.
```

3.2.2 ***Keywords Extraction.*** The keyword is another crucial factor of side information because it reflects the importance of the featured item. TF-IDF (term frequency-inverse document frequency)

[1] https://www.merriam-webster.com/
[2] https://www.wikidata.org/

is used for keyword extraction due to its efficiency [32]. It estimates the frequency of a word in one sentence over the maximum in a collection of sentences with the same relation type and assesses the importance of a word in one set of sentences. For relations with training sentences, all sentences are aggregated as one document d, and TF-IDF is implemented based on the document. Other models can also be used for keyword extraction.

3.3 Prototypical Network with Side Information Embedding

Instead of adding a softmax layer directly after encoders for relation classification, we use prototypical networks to compute a prototype for each relation after encoders because some works show that prototypical networks work well for few-shot learning [7, 34]. They are simpler and more efficient than other meta-learning algorithms, making them suitable for few-shot or zero-shot learning tasks. By comparing the distance between query sentences with prototypes for each relation, we can classify the relation. In this section, we describe the prototypical network model and its transformation with weighted side information embedding for zero-shot learning to detect new relations.

The main idea for the prototypical network is to compute a prototype representing each relation. Each prototype is the mean vector of embedded sentences belonging to one relation.

$$c_i = \frac{1}{N}\sum_{i=1}^{N} f_\phi(x_i) \tag{6}$$

where c_i represents the prototype for each relation r_i and f_ϕ is an embedding function, which is a CNN encoder in our model. Instead of concatenating all hypernyms and keywords directly after each prototype, we argue that not all keywords are of equal importance. To determine a more accurate representation for each relation, we calculate a weighted side information embedding for each relation. The equation of side information embedding si is as follows:

$$si = f(\frac{h_1 + h_2}{2}) \oplus f(k_1) \oplus \cdots \oplus f(k_n) \oplus K \tag{7}$$

$$K = \sum_{m-n}^{m} (\frac{\alpha_i}{\sum_{i=m-n}^{m} \alpha_i} f(k_i)) \tag{8}$$

where $f(\cdot)$ is a word embedding model, h_1 and h_2 are two hypernyms for name entities and k_i denotes the keyword. Symbol $\oplus$ is the concatenation operator, n is determined by exploration search, m is the total number of keywords and α_i is a calculated weight by:

$$\alpha_i = \frac{count(k,s)}{size(s)} \cdot log(\frac{N}{sentence(k,S)}) \tag{9}$$

where s is each instance and N is the number of instances in a relation. The final representation for each prototype with side information embedding ps_i can be expressed by:

$$ps_i = c_i \oplus si_i \tag{10}$$

The probabilities of the relations in $\mathfrak{R}$ for a query instance x is computed as follows:

$$p_\phi(y = ps_i|x) = \frac{exp(-d(f_\phi(x), ps_i))}{\sum_{ps_i'} exp(-d(f_\phi(x), ps_i^{'}))} \tag{11}$$

where $d(.)$ is Euclidean distance function as below:

$$d(f_\phi(x), ps_i) = \sqrt{\sum_{i=1}^{n}(ps_i - f_\phi(x))^2} \tag{12}$$

We use Euclidean distance instead of cosine similarity for distance calculation because previous work shows that Euclidean distance can improve performance substantially over cosine similarity [34]. We have not added any attention layer in our final model because (1) previous work shows there is little improvement on performance compared with vanilla prototypical networks [7]; (2) Ablation study in Section 4.4.3 shows there is no improvement on ZSLRC with attention layers.

For the zero-shot learning task, each relation is given the embedding for side information of the relation rather than a small number of labeled training sentences. We take the embedding of side information into a shared space to serve as the prototype for each relation. The core idea in traditional prototypical networks is to use an average embedding to represent a class [7, 34]. If there are no training data in that class, a high-level description of the class is used to represent that class. We modify prototypical networks to deal with both relations with training sentences and relations without training sentences. The difference between traditional prototypical networks and our proposed model is that they calculate the distance between the query sentence and prototype of each class to find the nearest one. Our proposed model first decides the query sentence is in a class with training data or the one without any training data based on a threshold. The reason is that finding the nearest distance directly based on all classes (with training data and without training data) is not fair for the class without training data because the high-level description is too general that it always has a longer distance compared with the classes which have training data.

We modify the prototypical network as follows: We first compare the distance between an input sentence with each prototype of known relations. The key mechanism for extracting new relations is that if the above distance is larger than a threshold, we consider the sentence has a new relation. Then we take the side information embedding of the input sentence and compare the distance between it with prototypes for new relations. Then we use a softmax layer to compute the probabilities for each new relation. The threshold selection is essential because it influences the decision of a relation type as an existing relation or a new relation. We implement a grid search to select the optimal threshold on the validation set. The entire framework of ZSLRC model to deal with a combination of known relations and new relations is described in Algorithm 2.

4 EXPERIMENTS

In this section, we conduct several experiments on two public datasets: NYT [33] and FewRel [9] to show that our proposed model outperforms other existing models on both a noisy dataset with a large number of training sentences and a clean dataset with few training sentences. We design experiments for generalized zero-shot learning tasks and provide a detailed analysis to show the effectiveness and advantages of our proposed model.

Algorithm 2: Algorithms for New Relation Extraction

```
Input  : prototype for each relation $c_i$, testing sentence $x$,
         threshold $t$.
Output : relationship $r$ of $x$.
Distance Calculation. $d(f_\phi(x), c_i)$.
Take $v = exp(-d(f_\phi(x), c_i))$.
if $v > t$ then
    Classification of known relations. $r = argmax(\frac{v}{\sum_{c_i'} v'})$
else
    Take side information embedding. $f_\phi(x)[SI\_DIM :]$
    Distance Calculation. $d(f_\phi(x)[SI\_DIM :], c_i)$.
    Take $v_{new} = exp(-d(f_\phi(x)[SI\_DIM :], c_i))$.
    Softmax of new relations. $r_{new} = argmax(\frac{v_{new}}{\sum_{c_i'} v_{new}'})$
```

4.1 Datasets and Evaluation Metrics

In our experiments, we evaluate our model over two widely used datasets: the NYT dataset [33] and FewRel [9] dataset. In the following, we describe each dataset in detail.

- ***NYT [33].*** The NYT dataset was generated by aligning Freebase relations with the New York Times corpus (NYT). There are 53 possible relationships in total. It is an unbalanced noisy dataset because all the relationships have a different number of sentences.
- ***FewRel [9].*** The FewRel dataset is a human-annotated few-shot RC dataset consisting of 80 types of relations, each of which has 700 instances.

To fairly compare the performance of our proposed model with other state-of-the-art models in supervised learning and few-shot learning tasks, we use the same training, validation and testing set of NYT dataset and same training and validation set of FewRel. We evaluate our proposed model on the validation set of FewRel because the test set is not available directly. In order to properly evaluate the performance of our proposed model in a zero-shot learning task, we re-split the above two public datasets for training, validation and testing set. Details of dataset re-splitting and experiment design are introduced in section 4.2. Note that we do not use any other clean, supervised dataset such as SemEval-2010 Task 8 (SemEval) because this dataset only contains 19 kinds of relations, which is less persuasive when re-splitting the dataset to evaluate the performance of our proposed model in zero-shot learning task [10].

The evaluation metrics adopted in this paper are the standard micro Accuracy (Acc.), Precision (Prec.), Recall (Rec.) and F1-score, similar to those used for the baseline.

4.2 Experiment Design

In a real-world scenario, there exist both kinds of relations with training instances and without any training instances. To make it simple and clear to understand, we call the relations with training instances known relations and the relations without any training instances new relations in the following discussion. To evaluate the effectiveness and robustness of our proposed model in a zero-shot learning task, we design the testing cases to contain different percentages (from 0% to 100% with a step of 10%) of new relations. Note that 0% means a thoroughly supervised learning or few-shot learning scenario, whereas 100% means a completely zero-shot learning scenario. The experiment design for zero-shot learning relation classification follows the criteria of zero-shot text classification; the different rates of unseen classes are used in testing cases [45].

NYT [33]. NYT is an unbalanced noisy dataset with 53 different relationships in total. We added initial training, validation and testing sets together and re-split the dataset into ten types of relations for the training pool. Each relation has over 10k sentences, and the rest relations are for the validation pool and testing pool. In the training pool, we take 10k sentences of each relationship for training, and the rest types of relations are used to validate and test known relations. In all, we have 100k sentences of 10 relationships in total for training, 13k sentences of known relations, and 5k sentences of new relations for validation and testing. For example, if a new relation *capital_of* is allocated to testing set, no *capital_of* sentences appear in training set.

FewRel [9]. FewRel dataset has 80 types of relations with 700 instances each. We re-split the dataset into 40 types of relations for training and 40 types of relations for testing. There are no overlapping relations among the training and testing sets. To evaluate our proposed model in a real-world scenario (a combination of known and new relations in the testing set), we take 300 instances from each relation type in the training set to make a testing pool containing known relations. In total, we have 40 relations, and each relation has 400 instances in the training pool, 40 known relations. Each relation has 300 instances in the testing pool, 40 new relations, and each relation has 700 instances in the testing pool.

4.3 Parameter Settings

For all the models, we use the pre-trained word embeddings with a 50-dimensional Glove model (6B tokens, 400K vocabulary) and a randomly initialized 5-dimensional position embedding on NYT corpus for initialization [30]. Both word embeddings and position embeddings are trainable during training. The number of feature maps in the convolutional layer is 800, and the side information embedding dimension is 300. We experimentally study the effects of two crucial parameters on our model, learning rate α and threshold t. We use a grid search to select the optimal learning rate α for SGD among $\{1e-1, 1e-2, 1e-3, 1e-4\}$ for minimizing the loss, the threshold t for determining a new relation among $\{2e-08, 7e-08, 2e-07, 7e-07\}$ on a validation set with 20% of new relations. The range of threshold is determined by the minimum and maximum values of e^{-d} on a validation set, where d is the Euclidean distance between query sentence and prototype for each relation. For other parameters, we follow the settings used in previous works so that our model can be fairly compared with these models [7, 43]. Table 1 shows parameters used in our experiment.

4.4 Results

4.4.1 ***Baseline Methods.*** We compare our proposed model to several state-of-the-arts models in both supervised learning and few-shot learning tasks. For a supervised learning task on the NYT dataset, we compare our model with **CDNN**, which first proposed

Table 1: Parameter Settings

Parameter	Value
Word Embedding Dimension d_w	50
Position Embedding Dimension d_p	5
Side Information Embedding Dimension d_{si}	300
Hidden Layer Dimension d_h	800
Convolutional Window Size n	3
Batch Size	1
Initial Learning Rate α	0.01
Weight Decay	10^{-5}
Threshold t	2e-08

Table 2: Results of different models on NYT (%). Our re-implementation is marked by $*$.

Model	Precision	Recall	F1
CDNN* [43]	46.4	52.7	45.8
REDN [19]	95.1	94.0	94.6
ZSLRC	**98.1**	**97.9**	**97.6**

Table 3: Ablation Results on NYT dataset (Accuracy%)

	10%	30%	50%	70%	90%
ZSLRC(HE)	88.94	70.57	52.12	33.87	15.48
ZSLRC(KE)	93.12	82.22	71.00	60.47	49.07
ZSLRC(SIE)	93.86	85.14	81.91	78.79	72.57
ZSLRC(WSIE)	**96.64**	**94.46**	**92.14**	**91.82**	**89.3**

the idea of position embedding [43]. The reason we choose this model to make the comparison is that we both use similar CNN encoders so that the improved performance of our model is not because of using any better encoders such as BERT [4]. The reported result for **CDNN** is our re-implementation on NYT because the source code is not available, and their original report is the evaluation on other datasets [43]. The reported result for the **REDN** is from the original published literature [19]. Note that **REDN** is a relation classification model using the given name entities, and we only copy the result of the single relation classification of this paper so that we could make a fair comparison. For few-shot learning task on FewRel dataset, we compare our model with **Meta Network**, **GNN**, **SNAIL**, **Proto**, **Proto-HATT** and **Proto-CATT(CNN)**. The six baselines above on the FewRel dataset are reported by [13], which are all current state-of-the-art FSL models. Note that the above FSL model Proto-HATT and our proposed model use the same pre-trained word embedding model 50-dimension GloVe, CNN encoders and same training parameters only except batch size and hidden layer dimension. For zero-shot learning, we compare our proposed model with the re-implemented **CDNN**, **REDN**, **Proto** and **Proto-HATT** on our re-splitted NYT and FewRel datasets to show the effectiveness and robustness of our proposed model.

4.4.2 *Results on NYT.*

Table 2 demonstrates that our proposed model achieves a substantial gain in precision, recall and F1-score over other baselines for the supervised learning task. We compare ZSLRC model with CDNN [43] as both models use a CNN encoder. The results show that ZSLRC achieves a significant performance improvement on precision, recall and F1-score. Our proposed ZSLRC also outperforms a recently proposed method (REDN) [19] by **3%** precision, **3.9%** recall and **3%** F1-score though REDN uses BERT encoder. This is important to note because BERT-based sentence encoders have significantly outperformed other sentence encoders including our proposed one-layer CNN based type [13].

The achieved performance improvement indicates that the proposed side information is competitively beneficial for relation classification. To evaluate our proposed model in a real-world scenario, we re-split the NYT dataset and use 40+ relations as new relations with no labeled training data. As is shown in Figure 5, 0% of new relations means it is a supervised learning task and all relations in the testing set have corresponding labeled training data. 100% of new relations means it is a conventional zero-shot learning task, and all relations in the testing set do not have any labeled training data. We compare the performance of our proposed model with CDNN [43] and REDN [19] as we vary the percentage of new relations in the testing set. As shown in Figure 5, the F1-score of both CDNN and REDN decrease when the percentage of new relations increase. This is because the model can not detect new relations and instead classifies the new relation as one of the existing relations in the training set. That is why the F1-score becomes zero when the new relation percentage is 100%. The F1-score of our proposed model ZSLRC only drops around 15% from a fully supervised case to a zero-shot case, indicating that our model is effective and sufficiently robust when dealing with new relations.

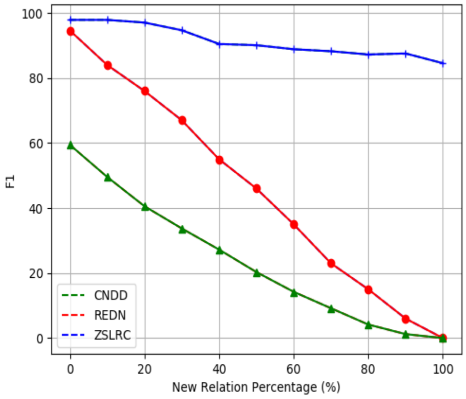


Figure 5: F1-score of ZSLRC when different proportions of new relations appear in NYT dataset.

To investigate the contribution of different side information embeddings in ZSLRC, we conduct an ablation study in zero-shot learning settings by adding each component, including hypernyms embedding (HE), keywords embedding(KE), side information embedding(SIE) and weighted side information embedding(WSIE). Table 3 shows the results of the ablation study different proportions of new relations in the testing set. We find out that all kinds of side information embedding help detect new relations. Only adding hypernyms embedding to the model can help detect new relation classes. However, the accuracy rate drops significantly from 88.94% in 10% of new relations to 15.48% in 90% of new relations. This is because hypernyms only represent the main categories for name entities and could help classify the relations roughly without training instances. Compared with hypernyms embedding, keywords embedding achieves much better performance because keywords (keywords extracted from training instances of seen class and synonyms of labels of unseen class) represent discriminative features of each instance, shorten the distance between query instance and prototype. Nevertheless, the performance of ZSLRC(KE) still drops considerably when the percentage of new relations increase. ZSLRC(SIE) achieves a significant accuracy performance improvement. Side information embedding is a combination of hypernyms embedding and keywords embedding. It represents high-level information of the instance, shortening the distance of instances with the same relation. As shown in Table 3, it is more robust when the percentage of the new relation class increases. Since we assume that not all side information is of equal importance, we also implement ZSLRC with weighted side information added to the model as introduced in Section 3.3. This model achieves the best performance. Besides the high accuracy performance with any proportions of new relations, it is also robust enough that it only drops 7.3% accuracy rate from 10% of new relations to 90% of new relations. Figure 6 indicates the accuracy improvement and robustness of weighted side information embedding. When the proportions of new relation increase, accuracy of ZSLRC with weighted side information embedding drops less than the other models.

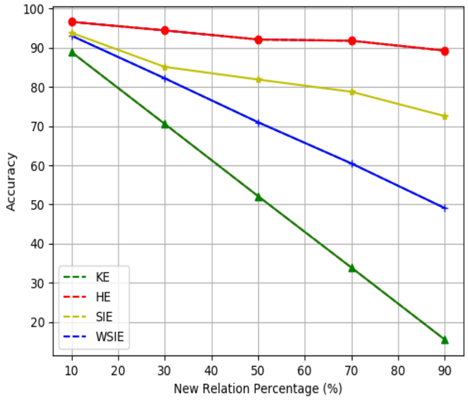


Figure 6: Ablation study of ZSLRC on NYT dataset.

4.4.3 ***Results on FewRel***. The evaluation results of few-shot learning on FewRel are shown in Table 4. Note that results with * are reported in [9]. The result of Proto-CATT model is copied from their original paper because of no public code [13]. We re-implement Proto and Proto-HATT with all parameters the same except hidden layer dimension. Both Proto-HATT and Proto-CATT are using CNN encoders and attention layers to help improve the performance. To fairly compare the effectiveness of side information embedding, we only compare our models with other state-of-the-art models using CNN encoders with attention layers. Each task is provided with a set of k labeled sentences from each of N classes that have not previously been trained upon. We conduct the experiments of N-way K-shot few-shot learning tasks following the method introduced in [29]. Table 4 shows that ZSLRC (without any attention layer) outperforms the other state-of-the-art models using multiple attention layers on several N-way K-shot tasks, especially for 1-shot cases. The accuracy of our proposed model on 5-way 1-shot and 10-way 1-shot tasks are 75.83% and 63.54%, which is 1.15% higher and 1.93% higher than the model Proto-HATT. Next, we investigate ZSLRC performance on N-way one-shot learning. Figure 7 demonstrates changes in accuracy as the number of ways changes in comparison with two state-of-the-art models. As the number of classes increases, the accuracy drops, but our proposed model has a slower dropping rate than other models. We conjecture that both the increased difficulty of a larger number of ways and the side information embedding we have proposed enables the ZSLRC to make more fine-grained decisions and is therefore more robust to the increased complexity introduced by more classes.

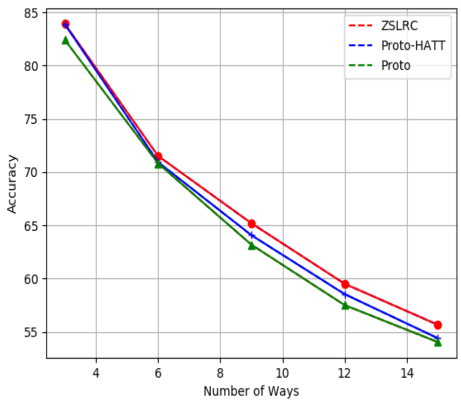


Figure 7: Accuracy of our proposed model in different N-way One-shot tasks.

To evaluate the effectiveness and robustness of ZSLRC in a generalized zero-shot learning task, we evaluate our models on the re-splitted FewRel dataset. To test the effectiveness and robustness of our proposed model, we compare our proposed model ZSLRC with Proto(CNN) and Proto-HATT(CNN) [7] in zero-shot settings described in Section 4.2. Figure 8 shows the performance of ZSLRC in a real world scenario with different percentages of new relations on re-splitted FewRel dataset. The accuracy of ZSLRC only drops from 97.3% to 86.8%, indicating the effectiveness and robustness of our proposed model for recognizing new relations in the real world. We show that zero-shot learning to new relation types is possible and we set the bar for future work on this task.

We also conduct an ablation study on FewRel dataset to learn the effectiveness of weighted side information embedding. Besides the models introduced in Section 4.4.2, we also implement a new

Table 4: Results of Accuracy Comparison Among Models (%)

Model	5 way 1 shot	5 way 5 shot	5 way 10 shot	10 way 1 shot	10 way 5 shot	10 way 10 shot
Meta Network*	64.46 ± 0.54	80.57 ± 0.48	-	53.96 ± 0.56	69.23 ± 0.52	-
GNN*	66.23 ± 0.75	81.28 ± 0.62	-	46.27 ± 0.80	64.02 ± 0.77	-
SNAIL*	67.29 ± 0.26	79.40 ± 0.22	-	53.28 ± 0.27	68.33 ± 0.25	-
Proto(CNN)	73.62 ± 0.20	85.78 ± 0.16	88.45 ± 0.10	60.96 ± 0.22	75.38 ± 0.19	78.71 ± 0.11
Proto-HATT(CNN)	74.68 ± 0.18	86.73 ± 0.12	89.64 ± 0.12	61.61 ± 0.16	77.04 ± 0.12	79.99 ± 0.11
Proto-CATT(CNN)	-	87.48 ± 0.12	89.28 ±0.08	-	77.46 ± 0.13	80.39 ± 0.14
ZSLRC(CNN)	**75.83±0.17**	**87.84±0.12**	**89.67±0.12**	**63.54±0.14**	**77.64±0.11**	**80.69±0.10**

Note that to fairly compare the performance of each model, we only compare the models with the same 50-dimension GloVe embedding and CNN encoders of the same parameters. Better results can be achieved through the BERT encoder.

Table 5: Ablation Results on FewRel dataset (%).

Model	5 way 1 shot	5 way 5 shot	5 way 10 shot	10 way 1 shot	10 way 5 shot	10 way 10 shot
Proto(CNN)	73.62 ± 0.20	85.57 ± 0.14	88.17 ± 0.10	62.22 ± 0.32	75.01 ± 0.16	78.50 ± 0.11
ZSLRC(HE)	75.66 ± 0.14	86.55 ± 0.13	88.98 ± 0.10	63.28 ± 0.20	76.58 ± 0.06	79.93 ± 0.05
ZSLRC(KE)	74.57 ± 0.08	86.70 ± 0.17	89.09 ± 0.11	62.39 ± 0.12	76.99 ±0.20	80.06 ± 0.09
ZSLRC(SIE)	75.56 ± 0.12	87.34 ± 0.14	89.17 ± 0.13	63.02 ± 0.15	77.16 ± 0.12	80.34 ± 0.10
ZSLRC(WSIE)	**75.83±0.17**	**87.84±0.12**	**89.67±0.12**	**63.54±0.14**	**77.64±0.11**	**80.69±0.10**
ZSLRC(WSIEA)	75.58 ± 0.15	87.16 ± 0.16	89.17 ± 0.15	62.85 ± 0.18	76.71 ± 0.14	80.18 ±0.11

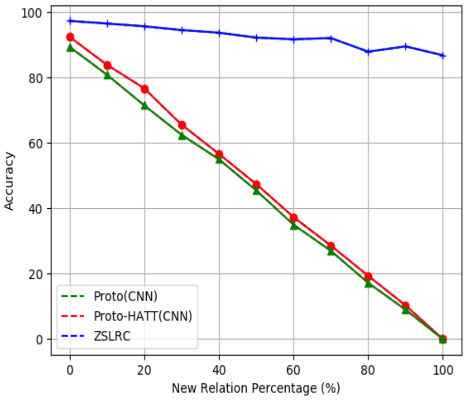


Figure 8: Accuracy of ZSLRC when different proportions of new relations appear in re-splitted FewRel dataset.

model with attention layers for weighted distance (WSIEA), to investigate the influence of the attention layer. Table 5 shows the results of ablation study. We can observe that all kinds of side information embedding contribute to the performance of ZSLRC. There is a big accuracy performance improvement when hypernyms embedding introduced in Section 3.2 is added to the model because hypernyms represent a general embedding for different name entities, which will decrease the variance from different word embeddings, leading to a shorter distance. Keyword embedding also contributes significantly to the performance, indicating the importance of keywords to side information embedding. Similar to the ablation result on NYT dataset as shown in Section 4.4.2, using side information embedding helps improve the performance and the model with weighted side information embedding achieves the best performance. We also added an attention layer built by three neural network layers and a softmax layer on top of each prototype to calculate linear separability based on the distribution of each prototype's sentence representations. However, there is no improvement of the attention layer. We conjecture that the weighted side information embedding has already captured each relation's vital feature. In this way, merely using side information embedding helps simplify the model's architecture, reducing the complexity of several neural network layers by attention mechanism.

5 CONCLUSION AND FUTURE WORK

We propose ZSLRC[3], a zero-shot learning relation classification framework based on modified prototypical networks. ZSLRC can detect new relations with no corresponding labeled data available for training. ZSLRC utilizes weighted side information constructed from labels, keywords and hypernyms of entities extracted from our proposed automatic hypernym extraction framework. We evaluate our model on supervised learning, few-shot learning and zero-shot learning tasks. The results demonstrate that our proposed ZSLRC outperforms other state-of-the-art models in all tasks. In addition, the results demonstrate the effectiveness and robustness of our proposed model. In future work, we plan to explore the following directions: (1) Due to the surprising performance improvement contributed by side information embedding, we will explore different ways to embed side information, leading to learning different representations of each prototype (relation). (2) We will explore using other popular sentence encoders such as BERT to improve the performance for relation classification.

[3] Implementation details can be accessed via: https://github.com/gjiaying/ZSLRC

REFERENCES

[1] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. 2015. Leveraging Linguistic Structure For Open Domain Information Extraction. In *Proceedings of the 53rd ACL and the 7th IJCNL*. 344–354.

[2] Matteo Bustreo, Jacopo Cavazza, and Vittorio Murino. 2019. Enhancing Visual Embeddings through Weakly Supervised Captioning for Zero-Shot Learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*.

[3] Lei Cui, Furu Wei, and Ming Zhou. 2018. Neural Open Information Extraction. In *Proceedings of the 56th Association for Computational Linguistics*. 407–413.

[4] J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

[5] Bowen Dong, Yuan Yao, Ruobing Xie, Tianyu Gao, Xu Han, Zhiyuan Liu, Fen Lin, Leyu Lin, and Maosong Sun. 2020. Meta-Information Guided Meta-Learning for Few-Shot Relation Classification. In *Pro. of the 28th ICCL*. 1594–1605.

[6] Cícero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying Relations by Ranking with Convolutional Neural Networks. In *Proc. of the 53rd Association for Computational Linguistics and the 7th IJCNLP*. 626–634.

[7] Tianyu Gao, Xu Han, Zhiyuan Liu, and Maosong Sun. 2019. Hybrid Attention-Based Prototypical Networks for Noisy Few-Shot Relation Classification. *Proceedings of the AAAI Conference on Artificial Intelligence* 33 (07 2019), 6407–6414.

[8] Tianyu Gao, Xu Han, Ruobing Xie, Zhiyuan Liu, Fen Lin, Leyu Lin, and Maosong Sun. 2020. Neural Snowball for Few-Shot Relation Learning. In *AAAI*.

[9] Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. FewRel: A Large-Scale Supervised Few-Shot Relation Classification Dataset with State-of-the-Art Evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 4803–4809.

[10] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *Proc. of the 5th Inter. Workshop on Semantic Evaluation*.

[11] Linmei Hu, Luhao Zhang, Chuan Shi, Liqiang Nie, Weili Guan, and Cheng Yang. 2019. Improving Distantly-Supervised Relation Extraction with Joint Label Embedding. In *Proc. of Conference on EMNLP and the 9th IJCNL*. 3821–3829.

[12] Lifu Huang, Heng Ji, Kyunghyun Cho, Ido Dagan, Sebastian Riedel, and Clare Voss. 2018. Zero-Shot Transfer Learning for Event Extraction. In *Proc. of the 56th Annual Meeting of the Association for Computational Linguistics*. 2160–2170.

[13] Bei Hui, Liang Liu, Jia Chen, Xue Zhou, and Yuhui Nian. 2020. Few-shot relation classification by context attention-based prototypical networks with BERT. *EURASIP Journal on Wireless Communications and Networking* 2020 (12 2020).

[14] Dat Huynh and Ehsan Elhamifar. 2020. Fine-Grained Generalized Zero-Shot Learning via Dense Attribute-Based Attention. In *Proceedings of the IEEE/CVF*.

[15] Shengbin Jia and Yang Xiang. 2020. Hybrid Neural Tagging Model for Open Relation Extraction. *arXiv: Computation and Language* (2020).

[16] Rohit Keshari, Richa Singh, and Mayank Vatsa. 2020. Generalized Zero-Shot Learning via Over-Complete Distribution. In *Proceedings of the IEEE/CVF*.

[17] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-Shot Relation Extraction via Reading Comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning*. 333–342.

[18] Aoxue Li, Weiran Huang, Xu Lan, Jiashi Feng, Zhenguo Li, and Liwei Wang. 2020. Boosting Few-Shot Learning With Adaptive Margin Loss. In *Proc. of the IEEE/CVF*.

[19] Cheng Li and Ye Tian. 2020. Downstream Model Design of Pre-trained Language Model for Relation Extraction Task. *ArXiv* (2020).

[20] Kai Li, Martin Renqiang Min, and Yun Fu. 2019. Rethinking Zero-Shot Learning: A Conditional Visual Classification Perspective. *2019 IEEE/CVF* (2019), 3582–3591.

[21] Yann Lifchitz, Yannis Avrithis, Sylvaine Picard, and Andrei Bursuc. 2019. Dense Classification and Implanting for Few-Shot Learning. In *Proceedings of the IEEE/CVF*.

[22] Xiao Ling and Daniel S. Weld. 2012. Fine-Grained Entity Recognition. In *Proc. of the Twenty-Sixth AAAI Conference on Artificial Intelligence*. AAAI Press, 94–100.

[23] ChunYang Liu, WenBo Sun, WenHan Chao, and WanXiang Che. 2013. Convolution Neural Network for Relation Extraction. 231–242.

[24] Colin Lockard, Prashant Shiralkar, Xin Luna Dong, and Hannaneh Hajishirzi. 2020. ZeroShotCeres: Zero-Shot Relation Extraction from Semi-Structured Webpages. In *Pro. of the 58th Association for Computational Linguistics*. 8105–8117.

[25] Diego Marcheggiani and Ivan Titov. 2016. Discrete-State Variational Autoencoders for Joint Discovery and Factorization of Relations. *Transactions of the Association for Computational Linguistics* 4 (2016), 231–244.

[26] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. 1990. Introduction to WordNet: An On-line Lexical Database*. *International Journal of Lexicography* 3, 4 (12 1990), 235–244.

[27] Thien Huu Nguyen and Ralph Grishman. 2015. Combining Neural Networks and Log-linear Models to Improve Relation Extraction. *CoRR* abs/1511.05926 (2015). arXiv:1511.05926

[28] Thien Huu Nguyen and Ralph Grishman. 2015. Relation Extraction: Perspective from Convolutional Neural Networks. In *Proc. of the 1st Workshop on Vector Space Modeling for Natural Language Processing*. 39–48.

[29] Alex Nichol, Joshua Achiam, and John Schulman. 2018. On First-Order Meta-Learning Algorithms. *ArXiv* (2018).

[30] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *EMNLP*. 1532–1543.

[31] Shafin Rahman, Salman Khan, and Nick Barnes. 2019. Transductive Learning for Zero-Shot Object Detection. In *Proceedings of the IEEE/CVF*.

[32] Juan Ramos. 2003. Using TF-IDF to determine word relevance in document queries. (01 2003).

[33] Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling Relations and Their Mentions without Labeled Text. In *MLKDD*, José Luis Balcázar, Francesco Bonchi, Aristides Gionis, and Michèle Sebag (Eds.). 148–163.

[34] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical Networks for Few-shot Learning. In *NIPS 30*. 4077–4087.

[35] Shikhar Vashishth, Rishabh Joshi, Sai Suman Prayaga, Chiranjib Bhattacharyya, and Partha Talukdar. 2018. RESIDE: Improving Distantly-Supervised Neural Relation Extraction using Side Information. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 1257–1266.

[36] Linlin Wang, Zhu Cao, Gerard de Melo, and Zhiyuan Liu. 2016. Relation Classification via Multi-Level Attention CNNs. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. 1298–1307.

[37] Ruidong Wu, Yuan Yao, Xu Han, Ruobing Xie, Zhiyuan Liu, Fen Lin, Leyu Lin, and Maosong Sun. 2019. Open Relation Extraction: Relational Knowledge Transfer from Supervised Data to Unsupervised Data. In *Proceedings of the 2019 Conference on EMNLP and the 9th IJCNP*. 219–228.

[38] Guo-Sen Xie, Li Liu, Xiaobo Jin, Fan Zhu, Zheng Zhang, Jie Qin, Yazhou Yao, and Ling Shao. 2019. Attentive Region Embedding Network for Zero-Shot Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[39] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. 2020. Few-Shot Learning via Embedding Adaptation With Set-to-Set Functions. In *Proceedings of the IEEE/CVF*.

[40] Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Multi-Level Matching and Aggregation Network for Few-Shot Relation Classification. In *Proc. of the 57th ACL*.

[41] Yunlong Yu, Zhong Ji, Jungong Han, and Zhongfei Zhang. 2020. Episode-Based Prototype Generating Network for Zero-Shot Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[42] J. Yuan, H. Guo, Z. Jin, H. Jin, X. Zhang, and J. Luo. 2017. One-shot learning for fine-grained relation extraction via convolutional siamese neural network. In *2017 IEEE International Conference on Big Data*. 2194–2199.

[43] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation Classification via Convolutional Deep Neural Network. In *Proceedings of COLING 2014*. 2335–2344.

[44] Dongxu Zhang and Dong Wang. 2015. Relation Classification via Recurrent Neural Network. *CoRR* abs/1508.01006 (2015). arXiv:1508.01006

[45] Jingqing Zhang, Piyawat Lertvittayakumjorn, and Yike Guo. 2019. Integrating Semantic Knowledge to Tackle Zero-shot Text Classification. In *Proc. of NAACL*.

[46] Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph Convolution over Pruned Dependency Trees Improves Relation Extraction. In *Proceedings of the 2018 Conference on EMNLP*. 2205–2215.

[47] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification. In *Proceedings of the 54th ACL*. 207–212.

[48] Hao Zhu, Yankai Lin, Zhiyuan Liu, Jie Fu, Tat-Seng Chua, and Maosong Sun. 2019. Graph Neural Networks with Generated Parameters for Relation Extraction. In *Proceedings of the 57th Association for Computational Linguistics*. 1331–1339.

[49] Jizhao Zhu, Jianzhong Qiao, Xinxiao Dai, and Xueqi Cheng. 2017. Relation Classification via Target-Concentrated Attention CNNs. 137–146.