Learning to Decouple Relations: Few-Shot Relation Classification with Entity-Guided Attention and Confusion-Aware Training

Yingyao Wang[1]*, **Junwei Bao**[2], **Guangyi Liu**[3], **Youzheng Wu**[2], **Xiaodong He**[2], **Bowen Zhou**[2], **Tiejun Zhao**[1]

[1]Harbin Institute of Technology, Harbin, China
[2]JD AI Research, Beijing, China
[3]The Chinese University of Hong Kong, Shenzhen, China
{baojunwei,wuyouzheng1,xiaodong.he,bowen.zhou}@jd.com
yywang@hit-mtlab.net,gy-liu@foxmail.com,tjzhao@hit.edu.cn

Abstract

This paper aims to enhance the few-shot relation classification especially for sentences that jointly describe multiple relations. Due to the fact that some relations usually keep high co-occurrence in the same context, previous few-shot relation classifiers struggle to distinguish them with few annotated instances. To alleviate the above *relation confusion problem*, we propose CTEG, a model equipped with two mechanisms to learn to decouple these easily-confused relations. On the one hand, an **E**ntity-**G**uided **A**ttention (EGA) mechanism, which leverages the syntactic relations and relative positions between each word and the specified entity pair, is introduced to guide the attention to filter out information causing confusion. On the other hand, a **C**onfusion-**A**ware **T**raining (CAT) method is proposed to explicitly learn to distinguish relations by playing a pushing-away game between classifying a sentence into a true relation and its confusing relation. Extensive experiments are conducted on the FewRel dataset, and the results show that our proposed model achieves comparable and even much better results to strong baselines in terms of accuracy. Furthermore, the ablation test and case study verify the effectiveness of our proposed EGA and CAT, especially in addressing the relation confusion problem.

1 Introduction

Relation classification (RC) aims to identify the relation between two specified entities in a sentence. Previous supervised approaches on this task heavily depend on human-annotated data, which limit their performance on classifying the relations with insufficient instances. Therefore, making the RC models capable of identifying relations with few training instances becomes a crucial challenge. Inspired by the success of few-shot learning methods in the computer vision community (Vinyals et al., 2016; Sung et al., 2017; Santoro et al., 2016) and some other natural language processing tasks (Chen et al., 2016; Qin et al., 2020; Zhou et al., 2019), Han et al. (2018) first introduce the few-shot learning to RC task and propose the FewRel dataset. Recently, many works focus on this task and achieve remarkable performance (Gao et al., 2019a; Snell et al., 2017; Ye and Ling, 2019).

Previous few-shot relation classifiers perform well on sentences with only one relation of a single entity pair. However, in real natural language, a sentence usually jointly describes multiple relations of different entity pairs. Since these relations usually keep high co-occurrence in the same context, previous few-shot RC models struggle to distinguish them with few annotated instances. For example, Table 1 shows three instances from the FewRel dataset, where each sentence describes multiple relations with corresponding keyphrases highlighted (colored) as evidence. When specified two entities (bold black) in the sentence, there is a great opportunity for the instance to be incorrectly categorized as a confusing relation (red) instead of the true relation (blue). Specifically, the first instance should be categorized as the true relation '*parents-child*' based on the given entity pair and natural language (NL) expression '*a daughter of*'. However, since it also includes the NL expression '*his wife*', it is probably misclassified into this confusing relation '*husband-wife*'. In this paper, we name it as a **relation confusion problem**.

*This work is done during the first author's internship in JD AI Research.

Proceedings of the 28th International Conference on Computational Linguistics, pages 5799–5809
Barcelona, Spain (Online), December 8-13, 2020

True Relation	Confusing Relation	Instance
parents-child	husband-wife	She was a daughter of prince Wilhelm of Baden and his wife ***princess Maria of Lichtenberg***, as well as an elder sister of ***prince Maximilian***.
husband-wife	uncle-nephew	He was the youngest son of ***Prescott Sheldon Bush*** and his wife ***Dorothy Walker Bush***, and the uncle of former president George W Bush.
uncle-nephew	parents-child	***Snowdon*** is the son of princess Margaret, countess of Snowdon, and the 1st earl of Snowdon, thus he is the nephew of ***queen Elizabeth ii***.

Table 1: Example sentences containing confusing relations. Their specified entities are marked as italics in bold. The **blue** and **red** words respectively correspond to true and confusing relations.

To address the relation confusion problem, it is crucial for a model to be aware of which NL expressions cause confusion and learn to avoid mapping the instance into its easily-confused relation. From these perspectives, we propose two assumptions. Firstly, *in a sentence, words that keep high relevance to the given entities are more important in expressing the true relation*. Secondly, *explicitly learning of mapping an instance into its confusing relation with augmented data in turn boosts a few-shot RC model on identifying the true relation*. Based on these assumptions, we propose CTEG, a few-shot RC model with two novel mechanisms: (1) An **E**ntity-**G**uided **A**ttention (EGA) encoder, which leverages the syntactic relations and relative positions between each word and the specified entity pair to softly select important information of words expressing the true relation and filter out the information causing confusion. (2) A **C**onfusion-**A**ware **T**raining (CAT) method, which explicitly learns to distinguish relations by playing a pushing-away game between classifying a sentence into a true relation and its confusing relation. In addition, inspired by the success of pre-trained language models, our approaches are based on BERT (Devlin et al., 2018), which has been proved effective especially for few-shot learning tasks.

Specifically, the backbone of the encoder of our model is a transformer equipped with the proposed **EGA** which guides the calculation of self-attention distributions by weighting the attention logits with entity-guided gates. The gates are used to measure the relevance between each word and the given two entities. Two types of information for each word are used to calculate its gate. One is the *relative position* (Zeng et al., 2015a) information, which is the relative distance between a word and an entity in the input sequence. The other is *syntactic relation* which is proposed in this paper, defined as the dependency relations between each word and the entities. Based on these information, the entity-guided gates in EGA are able to select those important words and control the contribution of each word in self-attention.

We also propose **CAT** to explicitly force the model to asynchronously learn the classification from an instance to its true relation and its confusing relation. After each training step, the CAT first selects those misclassified sentences, and regards the relations they are misclassified into as the confusing relations. After that, The CAT uses these misclassified instances and their confusing relations as augmented data to conduct an additional training process, which aims to learn the mapping between these instances into the confusing relations. Afterwards, the CAT adopts the KL divergence (Kullback and Leibler, 1951) to teach the model to distinguish the difference between the true and confusing relations, which benefits the true relation classification from the confusing relation identification.

The contributions of this paper are summarized as follows: (1) We propose an Entity-Guided Attention encoder, which can select crucial words and filter out NL expressions causing confusion based on their relevance to the specified entities. (2) We propose a Confusion-Aware Training process to enhance the model with the ability of distinguishing true and confusing relations. (3) We conduct extensive experiments on few-shot RC dataset FewRel, ans the results show that our model achieves comparable and even much better results to strong baselines. Furthermore, ablation and case studies verify the effectiveness of the proposed EGA and CAT, especially in addressing the relation confusion problem.

2 Methodology

2.1 EGA: Entity-Guided Attention Encoder

The inputs of our model include a sentence $S = w_1, ..., w_n$ with n words, and two pairs of integers $s_1 = (l_1, r_1)$ and $s_2 = (l_2, r_2)$ representing the start and end positions of the two specified entities.

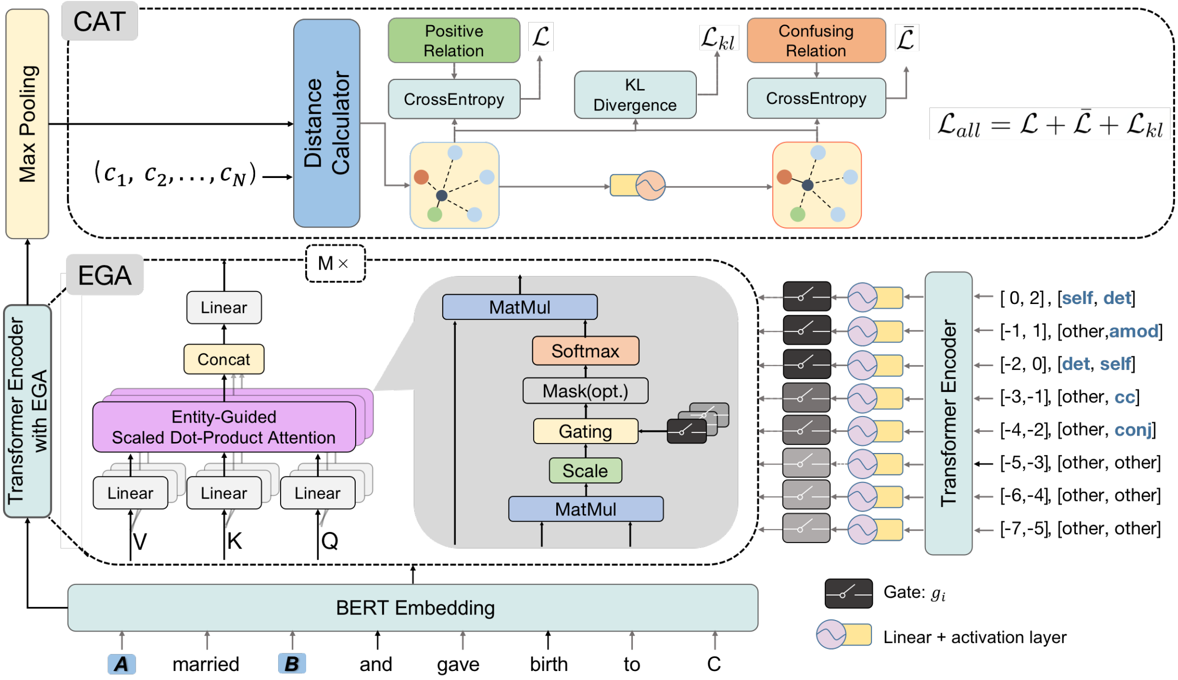
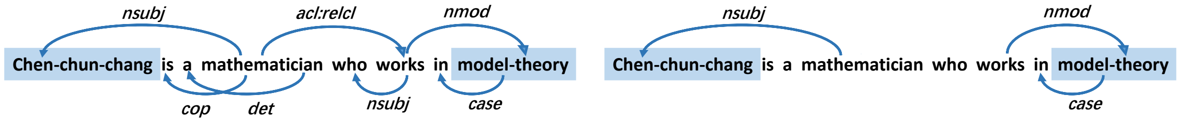


Figure 1: The framework of our model CTEG including **E**ntity-**G**uided **A**ttention (EGA) and **C**onfusion **A**ware **T**raining (CAT) mechanisms.


nsubj
acl:relcl
nmod
Chen-chun-chang is a mathematician who works in model-theory
cop
det
nsubj
case
nsubj
nmod
Chen-chun-chang is a mathematician who works in model-theory
case

(a) (b)

Figure 2: The dependency tree of a sentence and the paths used as syntactic relations in EGA.

Firstly, we convert the words into a sequence of vectors $\boldsymbol{e}_1^w, ..., \boldsymbol{e}_n^w$, using an embedding layer initialized by BERT. We then use two types of relevance information, i.e., relative position and syntactic relation, between each word and the specified entity pair to construct entity-guided gates for information selection.

Relative Position. Relative position information is typically used in relation classification task (Zeng et al., 2015a), which is defined as the relative distances pos_1 and pos_2 from the current word to the two specified entities in the sentence. The relative position information of the i-th word is represented as $\boldsymbol{e}_i^{pos} = [\boldsymbol{e}_i^{pos1}, \boldsymbol{e}_i^{pos2}]$, where $\boldsymbol{e}_i^{pos1}, \boldsymbol{e}_i^{pos2} \in \mathbb{R}^{d_{pos}}$ are the embeddings of pos_1 and pos_2.

Syntactic Relation Except for the relative position, we further introduce the syntactic relations to measure the relevance between each word and the specified entities. The syntactic relations are derived based on dependency parse trees, which are obtained from the Standford Parser[1]. For example, Figure 2(a) shows the original dependency tree of the sentence *"Chen-chun-chang is a mathematician who works in model-theory"*, where *"Chen-chun-chang"* and *"model-theory"* are entities. In this paper, we assume that words that directly connect to the given entities are more important in expressing the true relations. Therefore, dependency relations that connect the specified entities and other words are remained and the others are discarded, which derives a pruned dependency tree, as one shown in Figure 2(b). Based on the pruned dependency tree, each word in the sentence is assigned two tags $t_i = (t_{i,1}, t_{i,2})$ as the proposed **syntactic relations**. Taking the tag $t_{i,1}$ of word w_i which corresponds to the first entity as an example, if w_i is part of the first entity, the tag $t_{i,1}$ is assigned the value '*self*', and if w_i is directly connected to the first entity in the dependency tree, $t_{i,1}$ is assigned the dependency relation, e.g., '*nmod*'. In addition, if w_i is neither connected to nor part of the first entity, $t_{i,1}$ is assigned '*other*'. Based on the above strategy, the syntactic relations of the sentence in Figure 2 are shown in Table 2. Finally, the two dependency

[1] https://nlp.stanford.edu/software/lex-parser.shtml

Words	***chen chun chang***	*is*	*a*	*mathematician*	*who*	*works*	*in*	***model theory***
$t_{i,1}$	self	other	other	nsubj	other	other	other	other
$t_{i,2}$	other	other	other	other	other	nmod	case	self

Table 2: The syntactic relations corresponding to each word of the given sentence.

tags of each word $t_i = (t_{i,1}, t_{i,2})$ are converted into continuous vectors based on an embedding lookup operation, and then concatenated into a vector $\boldsymbol{e}_i^{syn} = [\boldsymbol{e}_i^{syn1}, \boldsymbol{e}_i^{syn2}]$, where $\boldsymbol{e}_i^{syn1}, \boldsymbol{e}_i^{syn2} \in \mathbb{R}^{d_{syn}}$.

Entity-Guided Gate The proposed EGA learns entity-guided gates $\boldsymbol{G} = (g_1, ..., g_n)$ for all words in a given sentence based on the above two types of information. Intuitively, if a word w_i is directly connected to the given entities in the dependency tree, the corresponding information tends to be more important. Specifically, the relative position embedding and the syntactic relation embedding are first concatenated into $\boldsymbol{e}_i^p = [\boldsymbol{e}_i^{pos}, \boldsymbol{e}_i^{syn}]$, where $\boldsymbol{e}_i^p \in \mathbb{R}^{2d_{pos}+2d_{syn}}$. We then adopt a transformer encoder (Vaswani et al., 2017) followed by a single layer feed-forward neural network (FNN) with $\texttt{sigmoid}(\cdot)$ activation to derive a sequence of entity-guided gates as follows:

$$(\boldsymbol{h}_1^p, ..., \boldsymbol{h}_i^p, ..., \boldsymbol{h}_n^p) = \texttt{TransEnc}(\boldsymbol{e}_1^p, ..., \boldsymbol{e}_i^p, ..., \boldsymbol{e}_n^p) \quad (1)$$

$$g_i = \texttt{sigmoid}(\boldsymbol{W}^g \boldsymbol{h}_i^p + \boldsymbol{b}^g) \quad (2)$$

Gated Self-Attention A pre-trained transformer encoder with M layers equipped with the proposed EGA is used to learn the representation for a sentence. The backbone of each layer is a self-attention layer, which calculates attention weights for word pairs in the sentence. We define self-attention weights of the tth-layer as $\boldsymbol{Att}^t$, and the corresponding hidden states of the sentence is represented as $\boldsymbol{H}^t$. To obtain the attention weights $\boldsymbol{Att}^t$, the scaled attention scores $\frac{\boldsymbol{Q}^t \boldsymbol{K}^{t\top}}{\sqrt{d_k}}$ is multiplied by the entity-guided gate $\boldsymbol{G}$ with broadcasting followed by a $\texttt{softmax}(\cdot)$ operation. The gated self-attention and the calculation of $\boldsymbol{H}^t$ are formalized as follows, where $\boldsymbol{W}_k$,$\boldsymbol{W}_q$,$\boldsymbol{W}_v$ are trainable parameters.

$$(\boldsymbol{Q}^t, \boldsymbol{K}^t, \boldsymbol{V}^t) = (\boldsymbol{W_q}, \boldsymbol{W_k}, \boldsymbol{W_v})\boldsymbol{H}^{t-1} \quad (3)$$

$$\boldsymbol{Att^t} = \texttt{softmax}(\frac{\boldsymbol{Q}^t \boldsymbol{K}^{t\top} \otimes \boldsymbol{G}}{\sqrt{d_k}}) \quad (4)$$

$$\boldsymbol{H}^t = \boldsymbol{Att}^t \boldsymbol{V}^t \quad (5)$$

Finally, vector $\boldsymbol{s}$ as the representation of the sentence is obtained based on Equation 6, where $\boldsymbol{H}^M$ represents the output of the M-th layer of the encoder.

$$\boldsymbol{s} = \texttt{maxpooling}(\boldsymbol{H}^M) \quad (6)$$

2.2 Classification

The classifier performs N-way-K-shot classification following few-shot learning paradigm and the prototypical network (Snell et al., 2017). Specifically, for a relation r_j where $j \in [1, N]$, K sentences are sampled from its instances firstly, and then these sentences are used to calculate the representation named prototype $\boldsymbol{c}_j$ of the relation. We define the representations of the K sentences as $\boldsymbol{s}_{j,1}^c, ..., \boldsymbol{s}_{j,K}{}^c$, and prototype $\boldsymbol{c}_j$ is calculated as follows:

$$\boldsymbol{c}_j = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{s}_{j,k}^c \quad (7)$$

Given the representation $\boldsymbol{s}^q$ of a sentence as query and the prototypes $(\boldsymbol{c}_1, ..., \boldsymbol{c}_N)$ of N relations, the model aims to classify $\boldsymbol{s}^q$ into one of the N candidate relations. We first obtain the distance distribution $\boldsymbol{\delta} = (\delta_1, ..., \delta_N)$ by calculating the Euclidean Distance between $\boldsymbol{s}^q$ and each prototype. Then, according to $\boldsymbol{\delta}$, the sentence will be classified into the nearest relation $\hat{r}$.

$$\boldsymbol{\delta} = (\|\boldsymbol{s}^q - \boldsymbol{c}_1\|^2, ..., \|\boldsymbol{s}^q - \boldsymbol{c}_N\|^2) \quad (8)$$

$$\hat{r} = \arg\min_{j}\,(\boldsymbol{\delta}) \tag{9}$$

To enable the classifier to learn confusing relations, we further project the distance distrubution $\boldsymbol{\delta}$ into $\bar{\boldsymbol{\delta}}$ via a FFN with a $\texttt{tanh}(\cdot)$ activation function defined as follows. The $\bar{\boldsymbol{\delta}}$ is used to predict the confusing relation defined as $\bar{r}$ during a confusion-aware training (CAT) stage which is introduced in Section 2.3.

$$\bar{\boldsymbol{\delta}} = \texttt{tanh}(\boldsymbol{W}^c\boldsymbol{\delta} + \boldsymbol{b}^c) \tag{10}$$

2.3 CAT: Confusion-Aware Training

The confusion-aware training is based on two asynchronous processes: *true relation identification* and *confusing relation identification*. During classifying a sentence, the former uses its true relation as the target, and the latter uses its confusing relation as the target. Specifically, given a sentence with its true relation as r, the training objective of the true relation identification is defined as:

$$\mathcal{L} = \texttt{CrossEntropy}(\texttt{OneHot}(r), \texttt{Softmax}(\boldsymbol{\delta})) \tag{11}$$

For the *confusing relation identification*, we first pick up those misclassified sentences after each training step of true relation identification, and use their prediction results as the targets. In formulation, assuming the sentence is misclassified into an incorrect $\bar{r}$, the objective function of the confusing relation identification $\bar{\mathcal{L}}$ is defined as:

$$\bar{\mathcal{L}} = \texttt{CrossEntropy}(\texttt{OneHot}(\bar{r}), \texttt{Softmax}(\bar{\boldsymbol{\delta}})) \tag{12}$$

Besides, the KL divergence is adopted as another objective function, which allows the model to learn to perform confusion decoupling. The KL divergence has the ability to push away the distance distribution $\boldsymbol{\delta}$ and $\bar{\boldsymbol{\delta}}$, and the formula is defined as follows:

$$\mathcal{L}_{kl} = -\texttt{KL}(\texttt{Softmax}(\boldsymbol{\delta}), \texttt{Softmax}(\bar{\boldsymbol{\delta}})) \tag{13}$$

Through minimizing $\mathcal{L}_{kl}$, the model is able to explicitly learn to distinguish relations by playing a pushing-away game between classifying a sentence into a true relation and its confusing relation. In other words, our model learns to explicitly decouple r and $\bar{r}$ for classification based on specified entities in a given sentence. It is worth noting that, only those misclassified sentences are used for updating the objective $\mathcal{L}_{kl}$. The final objective function of our model $\mathcal{L}_{all}$ is defined as $\mathcal{L}_{all} = \mathcal{L} + \bar{\mathcal{L}} + \mathcal{L}_{kl}$.

3 Experiments

In this section, we report our experiment results from the following four aspects. We first show the comparison results of our model CTEG and baselines on FewRel dataset in Section 3.3. We then demonstrate the effectiveness of the proposed entity-gated attention (EGA) and confusion-aware training (CAT) through the ablation studies in Section 3.4. In order to more intuitively and clearly show the role of EGA and CAT, we show their visualized examples in case study in Section 3.5. Furthermore, we verify that our model is capable of addressing the *relation confusion problem* to some extent in Section 3.6.

3.1 Implementation Details

Dataset The FewRel dataset (Han et al., 2018) contains 100 relations, which are split up into 64 for training, 16 for validation and 20 for testing. Each relation has 700 instances generated by distant supervision (Mintz et al., 2009). All the instances are annotated with a specified entity pair.

Settings The dimension of word embedding is set to 768 for consistency with the base model of BERT (Devlin et al., 2018). The max length of the input is set to 100. Following BERT, the layer number M of the transformer encoder with EGA is 12, and all parameters in it is initialized with the pretrained BERT model. The relative position and syntactic relation embedding dimensions are both set to 50, and the transformer encoder for obtaining entity-guided gates is set up with hidden size as 230, head number of self-attention as 2. In addition, the model is optimized by Adam algorithm with the learning rate and the weight decay as 1×10^{-5} and 1×10^{-6}, respectively.

Model	5-way-1-shot	5-way-5-shot	10-way-1-shot	10-way-5-shot
Proto (Han et al., 2018)	72.65 / 74.52	86.15 / 88.40	60.13 / 62.38	76.20 / 80.45
Proto-HATT (Gao et al., 2019a)	75.01 / −−	87.09 / 90.12	62.48 / −−	77.50 / 83.05
MLMAN (Ye and Ling, 2019)	78.85 / 82.98	88.32 / 92.66	67.54 / 73.59	79.44 / 87.29
BERT-PAIR (Gao et al., 2019b)	**85.66 / 88.32**	89.48 / 93.22	**76.84** / 80.63	81.76 / 87.02
CTEG (This work)	84.72 / 88.11	**92.52 / 95.25**	76.01 / **81.29**	**84.89 / 91.33**
w/o CAT	83.79	91.71	74.25	84.46
w/o EGA	72.94	86.71	61.88	77.03
w/ Pos	82.31	91.44	72.41	83.98
w/ Syn	83.61	91.78	73.94	84.89

Table 3: The main classification accuracy of baselines and our model are shown in **Validation / Test** format, where the test results are from FewRel public leaderboard[3]. All ablation results reported in this section are on the validation set.

3.2 Baselines

We implement four baselines on FewRel dataset: **Proto**, **Proto-HATT** (Gao et al., 2019a), **MLMAN** (Ye and Ling, 2019) and **BERT-PAIR** (Gao et al., 2019b). All the baselines are based on the few-shot learning framework. Specifically, for each training step, N relations are first sampled from the training set. For each of the above relation, K out of 700 instances are sampled to construct a supporting set, based on which a relation representation named *prototype* is calculated. Given an instance of the N relations to be classified, the models classify it by calculating the distances from it to N prototypes.

Proto & Proto-HATT Both of the two models adopt the convolutional neural network (CNN) as encoders. Proto calculates the prototype by averaging the representations of the K-instances in the supporting set, and classify the query using the Euclidean Distance. Differently, Proto-HATT further proposes a hybrid attention scheme which includes an instance-level attention and a feature-level attention, where the former is used to highlight the crucial support sentences in calculating the prototype, and the latter is to select more efficient features when calculating distances.

MLMAN Different from the Proto and Proto-HATT, MLMAN encodes each query and the supporting set in an interactive way by considering their matching information on multiple levels. At local level, the representations of an instance and a supporting set are matched following the sentence matching framework (Chen et al., 2017b) and aggregated by max and average pooling. At instance level, the matching degree is first calculated via a multi-layer perception (MLP). Then, taking the matching degrees as weights, the instances in a supporting set are aggregated to obtain the class prototype for final classification.

BERT-PAIR This model is based on the sentence classification model in BERT. The sentence to be classified is first paired with all the supporting instances, and then each pair is concatenated to a sequence. BERT takes this sequence as input and returns a relevance score, which is used to measure whether the given sentence expresses the same relation with the corresponding supporting instance.

3.3 Comparison with Baselines

Same as Proto, we set $N = 5, 10$ and $K = 1, 5$ for NvK few-shot learning. Average accuracy is used as the evaluation metric to evaluate the relation classification performance. The results in Table 3 show that our model EGA with CAT, named **CTEG**, outperforms the three strong baselines including Proto, Proto-HATT and MLMAN by a significant margin on all the settings. These improvements are mainly brought by our EGA and CAT, which help the model to classify those easily-confused instances into correct ones. In addition, applying pre-trained BERT also contributes to improving the performance. Compared with BERT-Pair, our CTEG achieves better result on 5v5, 10v1 and 10v5 settings and comparable results on 5v1 settings on the test set, while on the dev set our CTEG is slight lower than BERT-pair on 5v1 and 10v1 settings. We think that the lower performance on the dev set on 5v1 and 10v1 is due to the fact that BERT-Pair encodes two sentences together which benefits for information fusion, while models based on prototypical network rely on larger K supporting facts to get a better prototype.

3.4 Ablation Study

We conduct ablation study and show the results in Table 3. Firstly, we turn off the CAT of our full model, which is represented as "**w/o CAT**". In this case, the average results drops 0.43-1.76 point on the four settings. These drops indicate that the CAT has the ability to improve the classification performance. We then report three groups of results to verify the effectiveness of EGA. Specially, our model without EGA which only adopts the BERT as the encoder is denoted as "**w/o EGA**". It is worth noting that in this case, the model can not identify which words in a given sentence are entities. When the EGA is removed, the performance decreases obviously by 5.81-14.13 point. It is proved that the entity information is crucial for relation classification. Furthermore, "**w/ Pos**" means the entity-guided gates in EGA are calculated only using the relative position information, and "**w/ Syn**" only using the syntactic relation information. Compared with "**w/o EGA**", the results of these two groups are significantly improved. It shows that the syntactic relation information is more powerful than the relative position information, which means considering the dependency relations between each word and the specified entity pair boosts the performance of simply adopting traditional relative position information. In addition, it can be seen that the smaller size of the supporting set (1-shot v.s. 5-shot), the more absolute gain our CAT and EGA modules achieve. This phenomenon shows that our method performs well with fewer available supporting instances.

Model	CTEG w/ Syn	QGG	CTEG	FHG
5-way-5-shot	91.78 ± 0.22	90.64 ± 0.28	92.52 ± 0.31	90.57 ± 0.11

Table 4: Ablation Results on How, What, and When to Gate.

How to Gate The self-attention mechanism is used to update the representation of each *query* word by fusing the information of all *key* words in a given sentence. In this process, an attention score is calculated to leverage the contribution of each key word. In this paper, we propose to use gates to further adjust these attention scores. In our proposed EGA, each *entity-guided gate* reflects the relation between the key word and the specified entity pair, which is different for all key words. We also implement a baseline **QGG** with *query-guided gates*, where each gate reflects the relation between the key word and the query word. Specifically, the relation is modeled based on their syntactic relation if the key word is a specified entity, otherwise a '*other*' relation. The results of using these two kinds of gates in Table 4 shows that our model **CTEG w/ Syn** only modeling syntactic relations outperforms the **QGG** baseline, which further verifies that our EGA with entity-guided gates has the ability of effectively leveraging specified entity information to select input information.

What and When to Gate In our EGA, the entity-guided gates are used since the beginning of the encoding process by multiplying them with the self-attention scores in each transformer layer. It means that the information of the words is selected during learning the representation of them. Another baseline is to multiply gates with the **F**inal transformer **H**idden states of the words as **G**ating mechanism, which is defined as **FHG**. In this case, the information of all words has been fully fused before adopting gating mechanism for selection. As the results shown in Table 4, compared with our model **CTEG**, the accuracy of the **FHG** drops 1.95 point. The results indicate that earlier to gate the attention score during encoding as our EGA is more reasonable than only to adopt gating at the final hidden states.

3.5 Case Study

EGA visualized example The entity-guided gates in EGA are expected to emphasize the words which are more related to the true relation. To verify the effectiveness of EGA intuitively, we show the entity-guided gates heat map of a given instance in Figure 3. This instance is sampled from '*parent-children*' relation in the validation set of FewRel. As shown in the map, the words '*his mother is*' are given higher scores. Obviously, the three words are important for expressing the '*parent-children*' relation.

CAT visualized example In Figure 4, we visualize the distance distributions between the given sentence and its candidate relations. The four subfigures respectively show the distance distributions calcu-

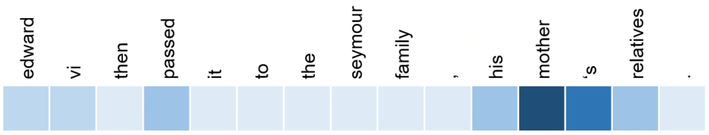


Figure 3: An example of the entity-guided gates of a given sentence.

lated by different models including our *true relation identification* and *confusing relation identification*. Among the five candidate relations, $R2$ in green is the true relation of the sentence, and R_1 in red is the confusing relation that the sentence is usually misclassified into. Each edge in the subfigures represents the distance from the sentence to the corresponding relation, and the solid edge indicates the nearest one. Specifically, (a) is the distances calculated by a randomly initialized network. (b) is the classification result of Proto, in this case, the query is misclassified into R_1. (c) and (d) are the final classification results of our CAT. The distance distribution between the query and the confusing relation calculated by our CAT is shown in (d), and it can be seen that the model succeeds in making the query closer to the confusing relation R_2 as we expected. After that, the distance distribution information is propagated to the true relation training by KL divergence, this operation is used to push the distance distribution of the true relation prediction away from the distribution of the confusing relation. As (c) shows, the sentence is pushed away from R_1 and get closer to the true relation R_2. This example validates our assumption that explicit learning of confusing relations facilitates the identification of true relations.

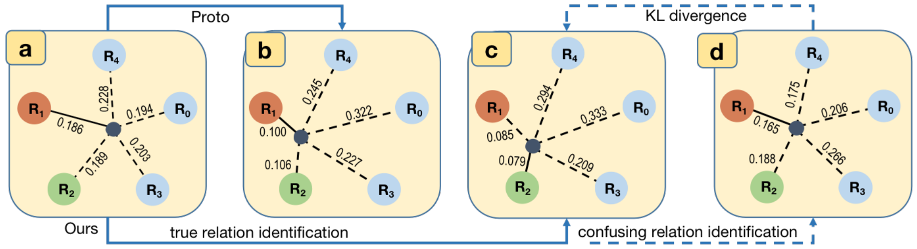


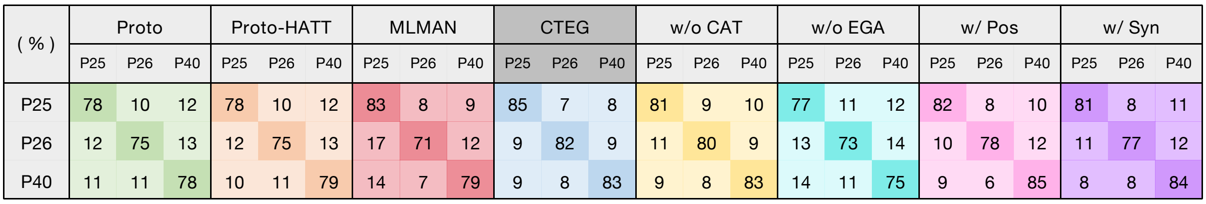
Figure 4: Distances between a given sentence and five candidate relations calculated by different models. where (a) is from a random initialized network, (b) is from the Proto network, (c) and (d) are respectively from our *true relation training* and *confusing relation training*.

3.6 Relation Confusion Problem

In this section, we discuss the effectiveness of our model on confusion decoupling and use the confusion matrices as our evaluation metric.

Confusing Relations Selection We first analyze the classification results of the baseline models Proto and Proto-HATT. Based on our statistics, we find three of the 16 relations in the FewRel validation set that are most easily confused with each other. Their relation indexes are ***P25***, ***P26*** and ***P40***, and the corresponding true relations are "***Parents-Child***","***Husband-Wife***" and "***Uncle-Nephew***". We test our model and the baseline models under the 5-way-5-shot configuration. For the three easily-confused relations, we respectively record the number of their sentences which are correctly classified and misclassified into the other two relations, and use the results to conduct the confusing matrices.

Improvement of Relation Confusion Problem As shown in Figure 5, we report the classification results of different models on the three confusing relations *P25*, *P26* and *P40*. In the confusion matrices, the horizontal axis represent the true relation of the sentences, and the vertical axis represent the classification results of these sentences by different models. For each matrix, supposing a given relation such as *P25* has X sentences to participate in the test, and the numbers of the sentences classified into *P25*, *P26* and *P40* are respectively a,b and c, than the elements in the first row of the matrix are calculated as $(a, b, c/X)$. Given a relation, we expect the models classify more its sentences into the true relation, and fewer its sentences into confusing relations. From this perspective, through comparing the confusion matrices of "CTEG" and the baseline models, it can be seen that our full model CTEG achieves



(%)	Proto			Proto-HATT			MLMAN			CTEG			w/o CAT			w/o EGA			w/ Pos			w/ Syn		
	P25	P26	P40	P25	P26	P40	P25	P26	P40	P25	P26	P40	P25	P26	P40	P25	P26	P40	P25	P26	P40	P25	P26	P40
P25	78	10	12	78	10	12	83	8	9	85	7	8	81	9	10	77	11	12	82	8	10	81	8	11
P26	12	75	13	12	75	13	17	71	12	9	82	9	11	80	9	13	73	14	10	78	12	11	77	12
P40	11	11	78	10	11	79	14	7	79	9	8	83	9	8	83	14	11	75	9	6	85	8	8	84

Figure 5: Confusion matrices of the three easily confused relations, where different colors represent the classification results of different models.

the best performance in identifying these easily confused relations. "w/o EGA" has the weakest ability to decouple the confusing relations, because it is not provided with any entity information to identify the true relation. Based on results of "w/o EGA", "w/ Pos" and "w/ Syn" we can see that both of the relative position and the syntax position bring significant improvements. In addition, compared with our full model, the performance of "w/o CAT" proves that the CAT help to decouple the confusing relations.

4 Related Work

Few-shot Relation Classification Relation classification (RC) aims to identify the semantic relation between two entities in a sentence, which is the basis of many natural language processing task, such as question answering (Yu et al., 2017) and knowledge graph completion (Shang et al., 2019). It has attracted more and more attention over past few years (Jia et al., 2019; Feng et al., 2018; Vinyals et al., 2018; Adel and Schütze, 2017; Yang et al., 2016a). Previous supervised approaches on this task heavily rely on labeled data for training, that limits their ability to classify the relations with insufficient instances. To address this problem, Han et al. (2018) first introduce few-shot learning to RC task, which has been proved effective in the computer vision community and has many applications (Vinyals et al., 2016; Sung et al., 2017; Santoro et al., 2016). Earlier works on few-shot RC are based on the widely used model prototypical network (Snell et al., 2017; Ye and Ling, 2019). Recently, the pre-trained language models (LM) has shown significant power in many natural language processing tasks. To this end, Gao et al. (2019c) adopt the most representative pre-trained LM BERT (Devlin et al., 2018) to few-shot RC, and their work shows that BERT brings significant improvements on classification performance. Furthermore, the approach proposed by Soares et al. (2019) are also based on BERT and achieve the state-of-art result on the few-shot RC task.

Syntactic Relation Previous RC models usually use the relative position information to identify which words are the entities in a sentence, e.g., Zeng et al. (2015b). In addition, the syntax information of the sentences is proved useful in many natural language processing tasks (Faleńska and Kuhn, 2019; Ma et al., 2020; Chen et al., 2017a). Inspired by Yang et al. (2016b), which adopt the dependency parse tree for RC (Ma et al., 2020), we also introduce the dependency relation as another type of position to emphasize the specific entities, and propose a novel application of the syntax positions.

5 Conclusions

In this paper, we propose CTEG equipped with two novel mechanisms, namely the Entity-Guided Attention (EGA) and the Confusion-Aware Training (CAT), to address the relation confusion problem in few-shot relation classification (RC). We conduct extensive experiments on benchmark dataset FewRel, and experiment results shows that our model achieves significant improvements on few-shot RC. Ablation studies verify the effectiveness of the proposed EGA and CAT mechanisms. Case study and further analysis demonstrate that our model has the ability of decoupling easily-confused relations.

Acknowledgments

We are grateful to the anonymous reviewers. The work of this paper is supported by the project of National High Technology Research and Development Program of China (No. 2018YFC0830700) and the National Key Research and Development Program of China under Grant 2018YFB2100802.

References

Heike Adel and Hinrich Schütze. 2017. Global normalization of convolutional neural networks for joint entity and relation classification. *arXiv preprint arXiv:1707.07719*.

Yun-Nung Chen, Dilek Hakkani-Tür, and Xiaodong He. 2016. Zero-shot learning of intent embeddings for expansion by convolutional deep structured semantic models. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6045–6049. IEEE.

Kehai Chen, Tiejun Zhao, Muyun Yang, and Lemao Liu. 2017a. Translation prediction with source dependency-based context representation. In *AAAI*.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017b. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada, July. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Agnieszka Faleńska and Jonas Kuhn. 2019. The (non-)utility of structural features in bilstm-based dependency parsers. pages 117–128, 01.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Tianyu Gao, Xu Han, Zhiyuan Liu, and Maosong Sun. 2019a. Hybrid attention-based prototypical networks for noisy few-shot relation classification. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence,(AAAI-19), New York, USA*.

Tianyu Gao, Xu Han, Hao Zhu, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2019b. FewRel 2.0: Towards more challenging few-shot relation classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6250–6255, Hong Kong, China, November. Association for Computational Linguistics.

Tianyu Gao, Xu Han, Hao Zhu, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2019c. Fewrel 2.0: Towards more challenging few-shot relation classification. *arXiv preprint arXiv:1910.07124*.

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 Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4803–4809. Association for Computational Linguistics.

Wei Jia, Dai Dai, Xinyan Xiao, and Hua Wu. 2019. ARNOR: Attention regularization based noise reduction for distant supervision relation classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1399–1408, Florence, Italy, July. Association for Computational Linguistics.

S. Kullback and R. A. Leibler. 1951. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Nianzu Ma, Sahisnu Mazumder, Hao Wang, and Bing Liu. 2020. Entity-aware dependency-based deep graph attention network for comparative preference classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5782–5788.

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.

Pengda Qin, Xin Wang, Wenhu Chen, Chunyun Zhang, Weiran Xu, and William Yang Wang. 2020. Generative adversarial zero-shot relational learning for knowledge graphs. *arXiv preprint arXiv:2001.02332*.

Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy P. Lillicrap. 2016. Meta-learning with memory-augmented neural networks. In Maria-Florina Balcan and Kilian Q. Weinberger, editors, *ICML*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1842–1850. JMLR.org.

Chao Shang, Yun Tang, Jing Huang, Jinbo Bi, Xiaodong He, and Bowen Zhou. 2019. End-to-end structure-aware convolutional networks for knowledge base completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3060–3067.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. 2017. Learning to compare: Relation network for few-shot learning. *CoRR*, abs/1711.06025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2018. Large-scale exploration of neural relation classification architectures. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2266–2277.

Yunlun Yang, Yunhai Tong, Shulei Ma, and Zhi-Hong Deng. 2016a. A position encoding convolutional neural network based on dependency tree for relation classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 65–74.

Yunlun Yang, Yunhai Tong, Shulei Ma, and Zhi-Hong Deng. 2016b. A position encoding convolutional neural network based on dependency tree for relation classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 65–74, Austin, Texas, November. Association for Computational Linguistics.

Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Multi-level matching and aggregation network for few-shot relation classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2872–2881, Florence, Italy, July. Association for Computational Linguistics.

Mo Yu, Wenpeng Yin, Kazi Saidul Hasan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2017. Improved neural relation detection for knowledge base question answering. *arXiv preprint arXiv:1704.06194*.

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

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015b. 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.

Fan Zhou, Chengtai Cao, Kunpeng Zhang, Goce Trajcevski, Ting Zhong, and Ji Geng. 2019. Meta-gnn: On few-shot node classification in graph meta-learning. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2357–2360.