# Better Semantic Representation: A Low-Shot Relation Extraction Method Based on Token-Generated Contributions

**Anonymous ACL submission**

## Abstract

In light of the era of information explosion, traditional relation extraction methods are in a bottleneck due to data limitations in the face of the constant emergence of new relation categories. Therefore the study of low-shot relation extraction in real scenarios is crucial. In the few-shot scenario, it is necessary to build up the model's ability to summarize the semantics of instances. In the zero-shot scenario, it is necessary to establish the label matching ability of the model. Although they need to establish different basic abilities of the model, the common point is that they all need to build excellent semantic representations in the end, which is ignored by the existing methods. In this paper, we propose a method (TGCRE) based on token-generated contribution to unify low-shot relation extraction by generating better semantic representations. Further, we propose a multi-level spatial semantic matching scheme in zero-shot scenarios, in order to solve the problem of the single matching pattern of existing methods. Experimental results show that our method outperforms previous robust baselines and achieves state-of-the-art performance.

## 1 Introduction

Relation extraction (RE) is an important basic task in natural language understanding. Traditional relation extraction relying on large-scale high-quality data has achieved excellent performance, but with the development of the times, high-quality data is consumed, and in the face of the emergence of various new relation categories that lack training data, the traditional methods are in a bottleneck. To cope with this situation, low-shot relation extraction has become a hot research topic. There are two main branches of low-shot relation extraction, namely the study of few-shot RE and zero-shot RE. The few-shot RE requires building the model's ability to summarize the semantics of instances, train the model's learning ability using a few labeled samples per class and quickly generalize it to classify new classes. At present few-shot RE approaches focus on how to summarize better semantic prototypes from a few illustrative examples(Snell et al., 2017), e.g. Gao et al. (2019a) et al. employ an attention mechanism to enhance the network's ability to generate prototypical representations. Han et al. (2021) et al. introduced a new approach based on supervised comparison learning in the hope that the model would learn good prototype representations, i.e., narrowing distances within classes while expanding distances between different classes. Another idea is to augment the FSRE model with knowledge from an external knowledge base. For example Wen et al. (2021) et al. introduced textual descriptions of entities and relations from Wikidata. Qu et al. (2020) et al. utilized the representation of global relation graphs. Yang et al. (2021) et al. utilized the intrinsic concept of entities. Zero-shot RE requires building the model's ability to match labels. The knowledge transfer capability of the model is trained and generalized to unseen relation categories by the labeled descriptions of the given relations. There are common solution paradigms such as question answering(Levy et al., 2017), textual entailment(Obamuyide and Vlachos, 2018) and semantic matching(Chen and Li, 2021). Despite the advanced performance achieved by semantic matching schemes, there are still some problems, the most representative of which is the single matching pattern, which causes the model to be negatively affected by irrelevant context when

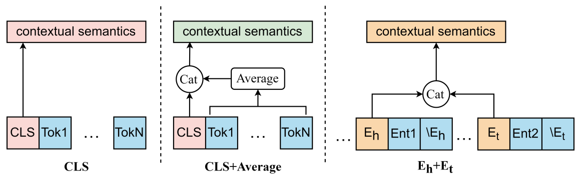


Figure 1: Semantic summarization methods

matching.

Since few-shot and zero-shot RE require the model to build different basic capabilities, current state-of-the-art methods can only be applied and learned to handle one scenario alone. However, what they have in common is that they ultimately need to construct good semantic representations, with few-shot RE requiring the semantic distance between the class prototype representation and its corresponding query instance to be reduced, and zero-shot RE requiring the model to summarize the semantic features of the different relation labels in a focused manner. Obviously, existing methods that rely only on the semantic summarization ability of special tokens inserted into sentences do not do this well, resulting in a model that does not summarize an optimal semantic representation. The existing methods for contextual semantic summarization are shown in Figure 1. See appendix E.1 for detailed analysis

For this reason, based on the commonalities between the above two we propose the method TGCRE, which utilizes and learns the token attributes inherent to each token in the sentence, i.e., the specific contribution each token makes to express the meaning of the sentence, to generate better semantic representations that unify the low-shot relational extraction. Moreover, in order to solve the problem of a single matching pattern in zero-shot RE, we propose a multi-level spatial semantic matching scheme. Label matching is performed by projecting semantic features to different vector spaces and synthesizing the matching scores from different perspectives. The contributions of this paper are summarized as follows:

1. We develop TGCRE, a low-shot relation extraction method for both zero-shot and few-shot tasks. Experiments demonstrate that our method outperforms previous baselines and achieves state-of-the-art performance in both zero-shot and few-shot tasks.

2. We propose a method for learning token attribute information, based on which a model is guided to understand the magnitude of the contribution of a token, and thus generate a better semantic representation of the context. To the best of our knowledge, we are the first to propose learning and using token attribute information for natural language understanding (NLU) tasks.

3. In the zero-shot RE task, we propose a multi-level spatial semantic matching scheme, which synthesizes the matching scores under multi-angle space to perform semantic matching and greatly improves the accuracy of semantic matching.

## 2 Related Work

**Zero-Shot Relation Extraction.** The task means to perform relation extraction on never-before-seen relation instances in the absence of annotated data for specific relation categories. Levy et al. (2017) et al. elucidated for the first time the concept of zero-sample learning for relation extraction by modeling the target task as a question-and-answer problem, and categorizing invisible classes by having the model answer a predefined question template. Obamuyide and Vlachos (2018) et al. modeled the target task as a textual entailment task, which identifies relation categories by determining whether the input sentences entail the corresponding relation descriptions, and fits well with the task definition of zero-sample learning. Sainz et al. (2021) et al. reformulate relation extraction as a problem of entailment, where a linguistic representation of relation labels is used to generate a hypothesis that is confirmed by a ready-made entailment engine. In the latest research, Chen and Li (2021) et al. use different projection functions for input text and relation description text respectively, transform both to the same semantic space, and based on this representation in the space defines relation extraction as a semantic matching task. Zhao et al. (2023a) et al. further proposed a fine-grained semantic matching method to reduce the impact of irrelevant context on matching accuracy. Wang et al. (2022) et al. use contrastive learning to train models that mitigate the prediction errors caused by similar relations and similar entities to the model. Recently, an even more difficult task, Zero-Shot Relation Triplet Extraction (ZSRTE)(Chia et al., 2022; Lv et al., 2023), has been proposed, which requires simultaneous extraction of both entities and relations, which greatly increases the task difficulty and further promotes the research on zero-shot relation extraction.

**Few-Shot Relation Extraction.** Few-shot learning is a challenging task when it relates to relation extraction. Few-shot RE aims to train a model by using only a small number of labeled samples and to improve the generalization ability of the model by utilizing unlabeled or weakly labeled data. When dealing with few-shot RE tasks, model training and testing are usually performed in a meta-learning manner(Mishra et al., 2017; Huisman et al., 2020; Hospedales et al., 2022). Snell et al. (2017) et al.

first proposed the use of prototypical networks for few-shot learning, Han et al. (2018) et al. further proposed a large-scale dataset, FewRel, to study relation extraction methods under few-shot learning. There has been an increase in the number of people involved in few-shot RE research. Gao et al. (2019a) et al. used an attention mechanism to facilitate the generation of better prototype representations from prototype networks. Ye and Ling (2019) et al. used CNN as an encoder and proposed a Multi-Level Matching and Aggregation Network for encoding query instances and class prototypes in an interactive interface. Gao et al. (2019b) et al. present a more challenging dataset, FewRel 2.0, in which they compute the similarity distance between a query instance and all supported instances. Han et al. (2021) et al. proposed representation modeling, prototype modeling and task difficulty modeling to solve difficult and simple few-shot extraction tasks. Recently, Liu et al. (2022) et al. proposed a simple direct additive method to introduce relation information, which proved that good relation information introduction is more effective than complex model structure. Li and Qian (2022) et al. proposed a model generation framework GM_GEN to achieve the optimal point on different N-way-K-shot tasks, separating the complexity of all the individual tasks from the complexity of the whole task space.

# 3 Preliminary

## 3.1 Encoding

**Sentence Encoding.** For any given input instance $I = \{x_1, x_2, \ldots, x_n\}$, the head entity $e_h^I$ and the tail entity $e_t^I$ are surrounded by the special symbols "#" and "@", respectively. We use the pre-trained language model BERT as a sentence encoder with encoded context features formulated as $\tilde{I} = \{h_1^I, h_2^I, \ldots, h_n^I\}$, and then extract the head entity feature $e_h^{\tilde{I}}$ and tail entity feature $e_t^{\tilde{I}}$ from the context features based on the locations of the specially tagged annotated entities using maximum pooling.

**Relation Description Encoding.** For any given relation description $d = \{d_1, d_2, \ldots, d_n\}$, we use an independently fixed sentence-BERT as a relation description encoder, following the work of Zhao et al. (2023a) et al., we extract the contextual features of the relation description $\tilde{d} = \{h_1^d, h_2^d, \ldots, h_n^d\}$ and the head entity description features $e_h^{\tilde{d}}$ and tail entity description feature $e_t^{\tilde{d}}$.

## 3.2 Token Attribution

For any given sentence, the tokens in the sentence work together and bear the responsibility of expressing the meaning of the sentence. However, each token makes a different specific contribution to the expression of the meaning of the sentence. For example, in the sentence "*I really like carrots.*", the contribution of "*really*" is obviously lower than that of "*like*". Without "*really*", the sentence can still convey the original meaning, but without "*like*", it is not clear whether I like carrots or hate them. We define this property as token attribution(Zhao et al., 2023b).

A measure of a token attribution can be defined by removing the token and observing the change in confidence that occurs when the model predicts the label of the instance.

$$g(x_i|I) = c(I) - c(I - x_i) \qquad (1)$$

where *c(I)* represents the confidence of the original sentence and $c(I - x_i)$ represents the confidence after removing the token $x_i$. $g(x_i|I)$ represents the attribution (contribution) of token $x_i$. When $g(x_i|I)$ is more than zero, i.e., $c(I) > c(I - x_i)$, it represents that the confidence of the model decreases after removing token $x_i$, which indicates that token $x_i$ has positive contribution in the sentence and can promote the expression of sentence meaning. Instead the token $x_i$ has a negative contribution in the sentence and can disrupt the model's predictions. Although the attribution of each token can be obtained in this way, it requires *n* forward computations, which is very inefficient and incurs a high computational overhead. Fortunately, computing the dot product of the corresponding embedding $h_i^I$ and gradient $\bigtriangledown_{x_i}$ for token $x_i$ can approximate the token attribution of $x_i$, so that the token attribution of all tokens can be obtained after only one forward-backward procedure. This approximation is proposed and applied in the interpretation methods of natural language classification models(Feng et al., 2018; Li et al., 2016; Arras et al., 2016). Thus, the method of measuring token attribution in practice can be formulated as:

$$attr\,(x_i|I) = \bigtriangledown_{x_i} \cdot h_i^I \qquad (2)$$

# 4 Methodology

In this section, we describe TGCRE in detail, and an overview of the methodology is shown in Figure

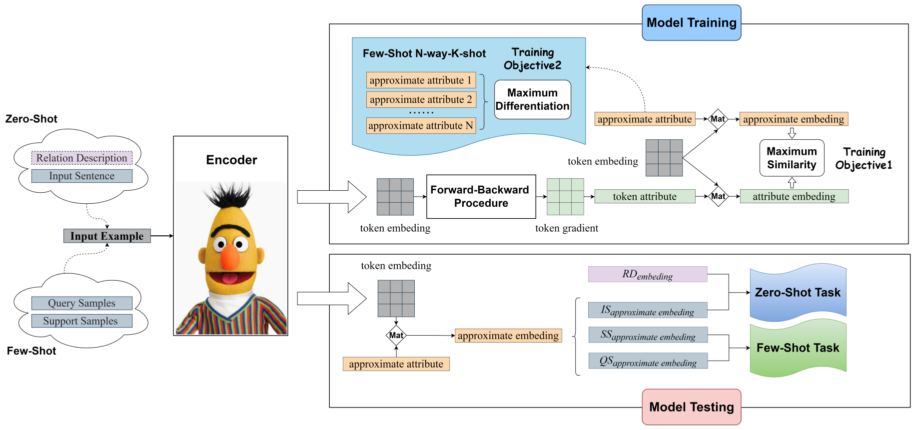


Figure 2: Model overview for TGCRE.

2. In the training phase of the model, the aim is to maximize the similarity between the *approximate attribute vector* and the *token attribution vector* and learn the attribute information of the tokens. In the testing phase, the learned knowledge of token attributes is used to guide the model to focus on the tokens with higher semantic contribution in the sentence, so as to generate better semantic representations for the subsequent zero/few-shot task. It is worth noting that the input example—Relation Description in the zero-shot setup uses an independently fixed encoder, Sentence-BERT, which is not labeled in Figure 2 for the sake of presentation simplicity.

### 4.1 Model Training

In the training phase, the goal is to learn information about the attributes of tokens so that the model has the ability to understand token contributions like a human. For the different inputs in the zero/few-shot setting, which we collectively refer to as input example *I*, which is encoded by the encoder to get the token embedding containing rich contextual semantics, i.e., $\tilde{I} = \left\{h_1^I, h_2^I, \ldots, h_n^I\right\}$.

**Forward-Backward Procedure.** In section 3.3, we introduced the first-order approximation for calculating token attribution, so we need a forward-backward procedure to obtain the gradient information for each token in the sentence. The backward process is straightforward, what matters is how the forward inference is performed so that tokens with larger contributions have more distinct gradients. We explore different forward inference approaches(See appendix E.2 for detailed analysis ) in this paper as follows:

(1) **Mean**: We treat the process of computing the mean of the token embeddings $\tilde{I}$ as forward propagation and the mean as the energy of backward propagation. In this pattern, there is no need to train any parameters other than those of the encoder. The advantage of this method is that it is relatively simple to implement.

$$forward : energy = MA\left(LSE\left(\tilde{I}\right)\right) \quad (3)$$

$$backward : BP\left(energy\right) \quad (4)$$

where $MA\left(\cdot\right)$ represents the mean function, *LSE* is *log-sum-exp* which gives better numerical stability and prevents the data from overflow and underflow problems during computation, and $BP\left(\cdot\right)$ which is the backward propagation of the model to obtain the gradient information.

(2) **Classification**: In order to obtain more reasonable gradient information, we insert a forward-backward procedure based on classification in the forward inference process of the whole method of TGCRE. This is done by training a classification function $cls\left(\cdot\right)$ and applying it to the word embedding $\tilde{I}$ so that the original word vector space is mapped into the relation vector space, obtaining the probability distribution of each relation corresponding to the input instance *I*. The loss is then calculated with the real label to get the energy as backward propagation. Compared to the Mean approach, this approach requires the training of an additional classification function, but the use of a supervised signal *y* allows the model to focus more on meaningful tokens and obtain more reasonable gradient information.

$$forward : energy = CEL\left(cls\left(LSE\left(\tilde{I}\right)\right), y\right) \quad (5)$$

$$backward : BP\left(energy\right) \quad (6)$$

where y represents the true label and $CEL(\cdot)$ represents the cross-entropy loss function, which is used to calculate the gap between the model's predictions and the true values.

**Normalization Token Attribution.** The gradient information $\bigtriangledown_{x_i}$ of all tokens can be obtained by one forward-backward procedure, which in turn can obtain all word attributes $\left|\bigtriangledown_{x_i} \cdot h_i^I\right|$. In order to visualize the specific degree of contribution of each token, it is necessary to normalize the token attributes to obtain the token attribute vector. The specific operation is shown below:

$$nta\left(x_i\right) = \frac{\left|attr\left(x_i|I\right)\right|}{\sum_{j=1}^{n}\left|attr\left(x_j|I\right)\right|} = \frac{\left|\bigtriangledown_{x_i} \cdot h_i^I\right|}{\sum_{j=1}^{n}\left|\bigtriangledown_{x_j} \cdot h_j^I\right|} \quad (7)$$

where $nta\left(x_1, x_2, \ldots, x_n\right)$ is the normalized token attribute vector.

**Training Objective1.** For the purpose of utilizing token attribute information and training the model for deeper understanding of natural language, a generalized approximate attribute vector *apa* that can learn token attribute information is proposed. We take maximizing the similarity between the approximate attribute vector natural language, a generalized approximate attribute vector *apa* and the token attribute vector *nta* as the training goal, so that *apa* is able to learn transferable token attribute knowledge, which in turn effectively guides the model to focus on the contributing tokens in the sentence and generate better semantic representations. First, the features of the token embedding $\tilde{I}$ are summarized based on the token attribute vector *nta*, and the attribute embedding is obtained by highlighting the positively contributing token features and ignoring the negatively contributing token features in the sentence. Secondly, the approximate attribute vector *apa* is also used to summarize the features of token embedding $\tilde{I}$, and approximate embedding is obtained. Finally, we use margin loss to optimize the training objective by iteratively training the model to shrink the similarity distance between attribute embedding and approximate embedding, and to increase the similarity between *apa* and *nta*, so as to continuously optimize the feature summarization ability of *apa*. The process can be formulated as:

$$\mathcal{L}_{sim} = max\left(0, 1 - cos(nta \cdot \tilde{I}, apa \cdot \tilde{I})\right) \quad (8)$$

**Training Objective2.** In the few-shot setting, we do not use a generalized approximate attribute vector due to the fewer number of relation categories that are restricted during the training process, but instead take the approach of setting a separate approximate attribute vector $apa_i$ for each relation category $r_i$. To prevent overfitting between the individual approximate attribute vectors, which causes most of the parameters to be invalidated, we introduce the second training objective — maximizing the differentiation between the groups of approximate attribute vectors. First, we compare the similarity between each two vectors $apa_i$ and $apa_j$, and then accumulate all the similarities to get the overall similarity score of the group of approximate attribute vectors, and use margin loss to reduce the value of the overall similarity score in differentiated training, thus preventing all the approximate attribute vectors from clustering in the same region in the vector space, and realizing the objective of differentiated training. The process can be formulated as:

$$\mathcal{L}_{Dif} = \max\left(0, \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\cos\left(apa_i, apa_j\right)}{N}\right) \quad (9)$$

### 4.2 Model Testing

In the testing phase, we use the trained approximate attribute vector *apa* to summarize the token embeddings and obtain the rich contextual semantics of the input examples for the subsequent few-shot RE task and zero-shot RE task. In the few-shot setting, the input examples include support samples and query samples, and the semantic representations after *apa* summarization are $SS_{approximate\ embeding}$ and $QS_{approximate\ embeding}$, respectively. In the zero-shot setting, the input examples consist of input sentence *I* and relation description *d*, where the summarized semantics of the *I* is represented as $IS_{approximate\ embeding}$, while the *d* is encoded using an independently fixed encoder that does not be summarized by the *apa*, and so the encoded semantics is represented as $RD_{embeding}$. It is worth mentioning that the semantic representations of the head and tail entities are extracted in token embeddings, and for the sake of brevity, this process is not shown in Figure 2.

**Zero-Shot RE Task.** In this paper, we define zero-shot RE as a semantic matching task, and in order to avoid the monotony of matching patterns, we propose a multi-level spatial semantic matching scheme. For the context

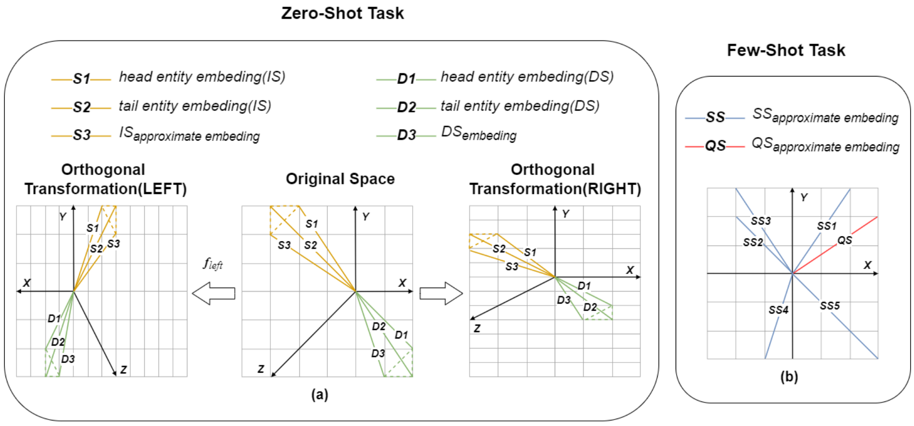


Figure 3: zero/few-shot task.

embedding $IS_{approximate\ embeding}$, head entity embedding $\tilde{e_h^I}$ and tail entity embedding $\tilde{e_t^I}$ of the input sentences in the given original vector space and the context embedding $RD_{embeding}$, head entity embedding $\tilde{e_h^d}$ and tail entity embedding $\tilde{e_t^d}$ of the relation descriptions, we define the embedding set of input sentences $SET_{IS} = \left\{\tilde{e_h^I}, \tilde{e_t^I}, IS_{approximate\ embeding}\right\}$ and the embedding set of relation descriptions $SET_{RD} = \left\{\tilde{e_h^d}, \tilde{e_t^d}, RD_{embeding}\right\}$. After that, we define the left orthogonal transform function $T_l\left(x, w_l\right)$ and the right orthogonal transform function $T_r\left(x, w_r\right)$, through which we can map the embedding set $SET_{IS}$ and the embedding set $SET_{RD}$ into different vector spaces.

$$SET_{IS}^l = T_l\left(SET_{IS}, w_l\right) \tag{10}$$

$$SET_{RD}^l = T_l\left(SET_{RD}, w_l\right) \tag{11}$$

$$SET_{IS}^r = T_r\left(SET_{IS}, w_r\right) \tag{12}$$

$$SET_{RD}^r = T_r\left(SET_{RD}, w_r\right) \tag{13}$$

where $w_l \in R^{3\times3}$, $w_r \in R^{h\times h}$ are trainable orthogonal matrices and $h$ is the hidden dimension of the encoder. As shown in Figure 3(a), we show a simple schematic of the embedding set transformation, although the real situation is much more complex than this. As can be seen from the figure, after the left (right) orthogonal transformation, $SET_{IS}$ and $SET_{RD}$ in the original space show different poses in different vector spaces, but the relative positions of the vectors in the embedding set are not changed, which ensures that their semantic similarities can be compared from different perspectives without changing the attributes of the original vector set.

We separately compute the semantic matching scores of the $SET_{IS}$ and $SET_{RD}$ in different vector spaces, and the sum of all the matching scores is used as the prediction scores of the input sentence $I$ and the relation description $d$.

$$p_z(I,d) = \alpha \cdot \cos\left(SET_{IS}^l, SET_{RD}^l\right) + \alpha \cdot \cos \left(SET_{IS}^r, SET_{RD}^r\right) + \beta \cdot \cos\left(SET_{IS}, SET_{RD}\right) \tag{14}$$

where $\alpha$ and $\beta$ are hyperparameters.

**Few-Shot RE Task.** In the N-way-K-shot setting, the context embedding is $SS_{approximate\ embeding}$ and $QS_{approximate\ embeding}$ for a given support set $S$ and query set $Q$ , respectively. We average the context embedding of each class in the support set $S$ to obtain a prototype representation $SS_i$ for each relation. As shown in Figure 3(b), the prototypical representation of each relation is randomly distributed in the vector space. In this paper, we use the cosine distance as the prediction score of the query instance for each class prototype and use the highest similarity as the final prediction.

$$P_f\left(S, Q\right) = \cos\left(SS_i, QS\right) \tag{15}$$

where $QS$ represents the context embedding $QS_{approximate\ embedding}$ of the query set.

### 4.3 Loss Function

In the zero-shot setting, in order to prevent the model overconfidence, we randomly sample the negative pairs to constrain the model, assuming that the prediction score of the positive pairs is $p_z\left(I, d_y\right)$, and that of the negative pairs is $p_z^i\left(I, d_i\right)$, then we require that the prediction score of the model's positive pairs is larger than that of the negative pairs, i.e., $p_z\left(I, d_y\right) -$

| Unseen | Method | Wiki-ZSL | | | FewRel | | |
|---|---|---|---|---|---|---|---|
| | | **Prec.** | **Rec.** | **F1** | **Prec.** | **Rec.** | **F1** |
| **m=5** | R-BERT | 39.22 | 43.27 | 41.15 | 42.19 | 48.61 | 45.17 |
| | ESIM | 48.58 | 47.74 | 48.16 | 56.27 | 58.44 | 57.33 |
| | ZS-BERT | 71.54 | 72.39 | 71.96 | 76.96 | 78.86 | 77.90 |
| | REPrompt | 70.66 | **83.75** | 76.63 | 90.15 | 88.50 | 89.30 |
| | RE-Matching | <u>79.84</u> | 78.58 | <u>79.19</u> | <u>91.48</u> | **90.84** | <u>91.16</u> |
| | **TGCRE** | **82.40** | <u>80.49</u> | **81.42** | **91.89** | <u>90.68</u> | **91.28** |
| **m=10** | R-BERT | 26.18 | 29.69 | 27.82 | 25.52 | 33.02 | 28.20 |
| | ESIM | 44.12 | 45.46 | 44.78 | 42.89 | 44.17 | 43.52 |
| | ZS-BERT | 60.51 | 60.98 | 60.74 | 56.92 | 57.59 | 57.25 |
| | REPrompt | 68.51 | **74.76** | 71.50 | 80.33 | 79.62 | 79.96 |
| | RE-Matching | <u>72.35</u> | <u>72.74</u> | <u>72.53</u> | <u>83.03</u> | <u>81.89</u> | <u>82.45</u> |
| | **TGCRE** | **74.61** | 72.07 | **73.30** | **86.23** | **85.11** | **85.66** |
| **m=15** | R-BERT | 17.31 | 18.82 | 18.03 | 16.95 | 19.37 | 18.08 |
| | ESIM | 27.31 | 29.62 | 28.42 | 29.15 | 31.59 | 30.32 |
| | ZS-BERT | 34.12 | 34.38 | 34.25 | 35.54 | 38.19 | 36.82 |
| | REPrompt | <u>63.69</u> | **67.93** | <u>65.74</u> | **74.33** | **72.51** | **73.40** |
| | RE-Matching | 62.35 | 62.34 | 62.33 | 73.11 | 70.36 | 71.69 |
| | **TGCRE** | **67.69** | <u>66.50</u> | **67.06** | <u>73.77</u> | <u>72.10</u> | <u>72.92</u> |

Table 1: Experimental results on the zero-shot task

$p_z^i(I, d_i) = \varphi > 0$, and the loss term is $\mathcal{L}_{lim} = max(0, \gamma - \varphi)$, where $\gamma > 0$ is a hyperparameter. To summarize, the total loss of the zero-shot RE is:

$$\mathcal{L}_z = \mathcal{L}_{sim} + \mathcal{L}_{lim} \tag{16}$$

In the few-shot setting, we use a cross-entropy loss function to optimize the gap between the model's prediction and the label, with a loss term of $\mathcal{L}_{cel} = CEL(p, y)$, where $p$ is the model's prediction and $y$ is the true label. To summarize, the total loss of the few-shot RE is:

$$\mathcal{L}_f = \mathcal{L}_{sim} + \mathcal{L}_{dif} + \mathcal{L}_{cel} \tag{17}$$

## 5 Experiments

In this section, we only show the main experimental results, and the experimental setup and detailed analysis are shown in the Appendix.

### 5.1 Experiments on Zero-Shot Relation Extraction

Table 1 summarizes the experimental results of our model with the baseline model on Wiki-ZSL and FewRel, where bold denotes the best score and underline denotes the second best score. In terms of F1 metrics, it can be seen that our model TGCRE significantly outperforms the other baselines, improving by 1.44% and 2.85% on the Wiki-ZSL and FewRel datasets, respectively. In terms of precision metrics, TGCRE shows excellent performance, substantially outperforming the existing baseline, which indicates that our model sufficiently learns the knowledge of token attribute and summarizes the semantic features of different relation labels in a focused manner. In terms of recall metrics, our model is slightly lower than REPrompt, but still performs reliably and outperforms the other baseline models. Overall, our model owes its state-of-the-art performance to token attribute knowledge and multilevel spatial semantic matching. RE-Matching has also achieved good results through fine-grained semantic matching due to display modeling of relational patterns.

### 5.2 Experiments on Few-Shot Relation Extraction

Table 2 summarizes the experimental results of our model with other models on the few-shot relation extraction task. As can be seen from the table, (1) our proposed TGCRE performs the best, indicating that our model is able to fully utilize the knowledge of token attribute to generate better semantic representations and effectively reduce the semantic distance between the class prototype representation and its corresponding query instance. (2) GM_GEN also achieves better performance by

| Method | 5-way-1-shot | 5-way-5-shot | 10-way-1-shot | 10-way-5-shot |
|---|---|---|---|---|
| | **validation/test** | **validation/test** | **validation/test** | **validation/test** |
| Proto-HATT | 75.01/— | 87.09/90.12 | 62.48/– – | 77.50/83.05 |
| MLMAN | 79.01/82.98 | 88.86/92.66 | 67.37/75.59 | 80.07/87.29 |
| BERT-PAIR | 85.66/88.32 | 89.48/93.22 | 76.84/80.63 | 81.76/87.02 |
| REGRAB | 87.95/90.30 | 92.54/94.25 | 80.26/84.09 | 86.72/89.93 |
| HCRP | 94.10/96.42 | 96.05/97.96 | 89.13/93.97 | 93.10/96.46 |
| SimpleFSRE | 96.21/96.63 | 97.07/97.93 | 93.38/94.94 | 95.11/96.39 |
| GM_GEN | <u>96.97/97.03</u> | <u>98.32/98.34</u> | <u>93.97/94.99</u> | <u>96.58/96.91</u> |
| TGCRE | **97.88/98.32** | **98.71/99.02** | **95.75/95.55** | **97.79/97.84** |

Table 2: Experimental results on the few-shot task

separating different N-way-K-shot tasks and allowing a single model to focus on a single task. We believe that it may be due to the "ONE-for-ONE" setting of GM_GEN that the model can focus on a specific task to generate semantic representations. (3) The model REGRAB, which uses external knowledge, did not achieve the expected results, a possible reason being that although external knowledge can bring additional reference information to the model, it can also introduce noise and limit the model's performance. (4) SimpleFSRE achieves good performance by introducing relational information through direct addition, again demonstrating that generating better semantic representations is often more important than complex network structures.

## 6 Ablation study

In order to understand the specific contribution of each component of the TGCRE model, we designed the following ablation experiments, and the results are shown in Table 3. When the token attribute vector is removed alone, i.e., the model is not allowed to learn the token attribute knowledge to summarize the contextual semantics, the model performance drops significantly. This suggests that token attribute can effectively guide the model to focus on important tokens and generate semantic representations containing rich contextual features. When removing the multi-level spatial semantic matching alone, the model performance also gets degraded, which shows that synthesizing the semantic matching scores under different vector spaces can improve the model performance and outperform the previous single matching pattern. When both of the above modules are removed at the same time, the model performance is severely impaired. From TGCRE (-attributue) and TGCRE (-both), it can be seen that the model performance is greatly impaired by removing the multi-level matching scheme on top of removing the token attribute vector, indicating that relying on the multi-level matching scheme alone can still allow the model to maintain excellent performance when there is no excellent semantic representation support.

| **Method** | **Prec.** | **Rec.** | **F1** |
|---|---|---|---|
| -attributue | 90.24 | 89.34 | 89.99 |
| -zj | 91.39 | 90.78 | 91.08 |
| -both | 88.98 | 87.19 | 88.06 |
| TGCRE | **91.89** | **90.68** | **91.28** |

Table 3: Ablation experiments on the FewRel dataset(unseen=5).

## 7 Conclusions

In this paper, we propose TGCRE, a low-shot relation extraction method based on token-generated contribution. The TGCRE summarizes instance features based on the specific contributions made by each token to generate better semantic representations that unify low-shot relation extraction. Specifically, TGCRE learns knowledge of token attributes by training approximate attribute vector, which guides the model to focus on tokens that contribute significantly to sentence expression. Moreover, in the zero-shot scenario, we propose a multi-level spatial semantic matching scheme that synthesizes the matching scores from different perspectives for label matching and greatly improves the matching accuracy. Extensive experiments have proved the effectiveness of our method, achieving state-of-the-art performance.

## Limitations

The token attribute information has been shown to facilitate the model in generating better semantic representations, and although we propose two approaches for generating gradient information in the paper (Mean, Classification), this is still not the optimal choice. Exploring richer gradient generation approaches that motivate models to better utilize token attribute information is a promising direction that will be the focus of our future work.

## References


Leila Arras, Franziska Horn, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. 2016. Explaining predictions of non-linear classifiers in NLP. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 1–7, Berlin, Germany. Association for Computational Linguistics.

Chih-Yao Chen and Cheng-Te Li. 2021. ZS-BERT: Towards zero-shot relation extraction with attribute representation learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3470–3479, Online. Association for Computational Linguistics.

Yew Ken Chia, Lidong Bing, Soujanya Poria, and Luo Si. 2022. RelationPrompt: Leveraging prompts to generate synthetic data for zero-shot relation triplet extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 45–57, Dublin, Ireland. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber. 2018. Pathologies of neural models make interpretations difficult. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium. Association for Computational Linguistics.

Tianyu Gao, Xu Han, Zhiyuan Liu, and Maosong Sun. 2019a. Hybrid attention-based prototypical networks for noisy few-shot relation classification. AAAI'19/IAAI'19/EAAI'19. AAAI Press.

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. Association for Computational Linguistics.

Jiale Han, Bo Cheng, and Wei Lu. 2021. Exploring task difficulty for few-shot relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2605–2616, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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*, pages 4803–4809, Brussels, Belgium. Association for Computational Linguistics.

Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. 2022. Meta-learning in neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5149–5169.

Mike Huisman, Jan N. van Rijn, and Aske Plaat. 2020. A survey of deep meta-learning. *Artificial Intelligence Review*, 54:4483 – 4541.

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 (CoNLL 2017)*, pages 333–342, Vancouver, Canada. Association for Computational Linguistics.

Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. 2016. Visualizing and understanding neural models in NLP. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 681–691, San Diego, California. Association for Computational Linguistics.

Wanli Li and Tieyun Qian. 2022. Graph-based model generation for few-shot relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 62–71, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yang Liu, Jinpeng Hu, Xiang Wan, and Tsung-Hui Chang. 2022. A simple yet effective relation information guided approach for few-shot relation extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 757–763, Dublin, Ireland. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Bo Lv, Xin Liu, Shaojie Dai, Nayu Liu, Fan Yang, Ping Luo, and Yue Yu. 2023. DSP: Discriminative soft prompts for zero-shot entity and relation extraction. In Findings of the Association for Computational Linguistics: ACL 2023, pages 5491–5505, Toronto, Canada. Association for Computational Linguistics.

Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and P. Abbeel. 2017. A simple neural attentive meta-learner. In *International Conference on Learning Representations*.

Abiola Obamuyide and Andreas Vlachos. 2018. Zero-shot relation classification as textual entailment. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 72–78, Brussels, Belgium. Association for Computational Linguistics.

Meng Qu, Tianyu Gao, Louis-Pascal Xhonneux, and Jian Tang. 2020. Few-shot relation extraction via bayesian meta-learning on relation graphs. *ArXiv*, abs/2007.02387.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Oscar Sainz, Oier Lopez de Lacalle, Gorka Labaka, Ander Barrena, and Eneko Agirre. 2021. Label verbalization and entailment for effective zero and few-shot relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1199–1212, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jake Snell, Kevin Swersky, and Richard S. Zemel. 2017. Prototypical networks for few-shot learning. In *Neural Information Processing Systems*.

Shusen Wang, Bosen Zhang, Yajing Xu, Yanan Wu, and Bo Xiao. 2022. RCL: Relation contrastive learning for zero-shot relation extraction. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2456–2468, Seattle, United States. Association for Computational Linguistics.

Wen Wen, Yongbin Liu, Chunping Ouyang, Qiang Lin, and Tonglee Chung. 2021. Enhanced prototypical network for few-shot relation extraction. *Information Processing  Management*, 58(4):102596.

Shanchan Wu and Yifan He. 2019. Enriching pre-trained language model with entity information for relation classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 2361–2364, New York, NY, USA. Association for Computing Machinery.

Shan Yang, Yongfei Zhang, Guanglin Niu, Qinghua Zhao, and Shiliang Pu. 2021. Entity concept-enhanced few-shot relation extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 987–991, Online. Association for Computational Linguistics.

Zhiquan Ye and Zhenhua Ling. 2019. Multi-level matching and aggregation network for few-shot relation classification. In *Annual Meeting of the Association for Computational Linguistics*.

Jun Zhao, WenYu Zhan, Xin Zhao, Qi Zhang, Tao Gui, Zhongyu Wei, Junzhe Wang, Minlong Peng, and Mingming Sun. 2023a. RE-matching: A fine-grained semantic matching method for zero-shot relation extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6680–6691, Toronto, Canada. Association for Computational Linguistics.

Jun Zhao, Xin Zhao, WenYu Zhan, Qi Zhang, Tao Gui, Zhongyu Wei, Yun Wen Chen, Xiang Gao, and Xuanjing Huang. 2023b. Open set relation extraction via unknown-aware training. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9453–9467, Toronto, Canada. Association for Computational Linguistics.

## A Task Formulation

**Few-Shot RE.** In resource-poor few-sample scenarios, the purpose of few-shot relation extraction is to train the model's triplet extraction capability using only a small number of training samples when there are not a large number of labeled samples in the candidate class, usually with the number of samples specified in an N-way-K-shot setting. Specifically, there is a support set *S* and a query set *Q* in different N-way-K-shot tasks, respectively. *S* contains N randomly sampled relation categories $r \in \boldsymbol{R}_s$ and each class *r* corresponds to K labeled instances $s_i$ used for training. *Q* contains *m* (custom hyperparameters) query instances $q_i$ for testing. The goal of the few-shot RE task is to train the model's learning ability by supporting instances $s_i$ so that the model can quickly adapt and deal with similar types of tasks, rather than just a single classification task. Finally, the learning capability of the model is verified using instances $q_i$ in the query set *Q*, predicting to which of the categories *r* in $R_s$ that $q_i$ belongs. Formally, this can be formulated as:

$$S \xrightarrow{\text{train}} M(LB) \xleftarrow{\text{validation}} Q \tag{18}$$

where *M(LB)* represents the learning capacity learned by the model.

**Zero-Shot RE.** In zero-sample scenarios where no data resources are available, zero-shot RE aims to use existing well-labeled datasets to train the model's triple-extraction capability and then apply it to extract the relations of entity pairs from new unseen data. Specifically, each relation $r \in \boldsymbol{R}$ in the dataset corresponds to a relation description $d \in \boldsymbol{D}$. A model is trained to measure the distance between sentence instances $I$ and relation descriptions $D$, and to predict to which type $r$ in $R$ that $I$ belongs. The goal of zero-shot RE is to use relation-visible data $Y_s$ to train the knowledge transfer capability of the model, allowing the model to use past knowledge to infer and recognize new things that have not been seen before. Ultimately, relation-invisible data $Y_u$ is used to validate the model's knowledge transfer capability. Formally, this can be formulated as:

$$Y_s \xrightarrow{\text{train}} M(KG) \xleftarrow{\text{validation}} Y_u \tag{19}$$

where *M(KG)* represents the knowledge transfer capability learned by the model and $Y_s \cap Y_u = \emptyset$.

## B Datasets

We evaluated our method on two popular datasets in low-shot RE. The FewRel dataset is used in the few-shot RE task, and the FewRel and Wiki-ZSL datasets are used in the zero-shot RE task.

**FewRel** dataset consists of 70,000 sentences from 100 relations on Wikipedia, annotated by crowd-funding workers. The standard FewRel follows the setup of training/validation/testing sets corresponding to 64/16/20 relation categories, where the training and validation sets are publicly accessible, whereas the testing set is not.

**Wiki-ZSL** dataset contains 113 relations and 94,383 instances from Wikipedia, completed by remote supervised annotation. The dataset is divided into three subsets: training set/validation set/test set, corresponding to 98/5/10 relation categories, respectively.

## C Baseline Models

In order to evaluate the effectiveness of our method, we compare TGCRE with state-of-the-art methods in the few-shot RE and zero-shot RE tasks, respectively, selecting a representative number of models from recent years.

For the few-shot RE, the models include Proto-HATT(Gao et al., 2019a), MLMAN(Ye and Ling, 2019), BERT-PAIR(Gao et al., 2019b), RE-GRAB(Qu et al., 2020), HCRP(Han et al., 2021), SimpleFSRE(Liu et al., 2022), and GM_GEN(Li and Qian, 2022). For zero-shot RE, the models include R-BERT(Wu and He, 2019), ESIM(Levy et al., 2017), ZS-BERT(Chen and Li, 2021), RE-Prompt(Chia et al., 2022), and RE-Matching(Zhao et al., 2023a).

## D Experimental settings

Following existing methods, we use Bert-base(Devlin et al., 2019) as an encoder for the input sentences. In particular, we employ a separate fixed sentence-Bert(Reimers and Gurevych, 2019) for the relation descriptions as an encoder, with the aim of reducing the computational overhead.

In the zero-shot RE task, the learning rate is set to 2e-6, batchsize is set to 16, and 10 epochs are trained. We randomly choose $m \in \{5, 10, 15\}$ relations as visible relations in the test set and consider the rest as visible relations in the training set. In this paper, we randomly repeat the relation category selection five times and report the average results under different selections to ensure the reliability of the experimental results.

In the few-shot RE task, the learning rate is set to 1e-5, the batchsize is set to 2, and the number of training iterations and validation iterations are set to 30,000 and 1,000, respectively. Following the official evaluation setup, we use 5-way-1-shot, 5-way-5-shot, 10-way-1-shot, and 10-way-5-shot to measure the performance of the model on the validation and test sets.

AdamW(Loshchilov and Hutter, 2017) is used as an optimizer in both the above tasks. In this paper, the IDE used for the experiments is Pycharm 2021 Professional Edition. PyTorch version 1.9.1; CUDA version 11.7. model training and inference were performed on an NVIDIA A100-SMX with 40GB of GPU memory and 16GB of CPU memory.

## E Case Study

### E.1 Analysis of different semantic summarization approaches

In order to compare the advantages and disadvantages of each semantic summarization approach, we designed the following comparison experiments, and the results are shown in Table 4. We take the FewRel dataset as an example and use TGCRE as the base model for zero-shot relation extraction using different semantic summarization approaches. From the experimental results, it can be seen that the semantic summarization approach based on token attributes proposed in this paper achieves the best performance in all three metrics, which is

| Method | Prec. | Rec. | F1 |
|---|---|---|---|
| CLS | 91.38 | 90.47 | 90.92 |
| CLS+Avg | 89.56 | 88.44 | 88.99 |
| $\mathrm{E}_h + E_t$ | 90.24 | 89.34 | 89.99 |
| Attribute | **91.89** | **90.68** | **91.28** |

Table 4: Comparison of different semantic summarization approaches.

superior to previous approaches based on special tokens. In particular, *CLS+Avg* achieves only 88.99 and $E_h + E_t$ up to 89.99 in terms of F1 metrics, which suggests that they do not seem to achieve the desired results in an unsupervised task that lacks supervised signals. Instead, the use of the most simple [CLS] as an embedding token for semantic summarization reached 90.92, just below our proposed approach.

### E.2 Analysis of different forward-backward procedures

In order to understand the impact of our proposed two forward-backward procedures, *Mean* and *Classification*, on the performance of the model, we set up relevant experiments by randomly sampling the set of invisible relations five times with unseen=5. The experimental results are shown in Table 5. We observe the counterfactual that the *Classification* method based on supervised labeling is actually lower than the simple *Mean* method, although there is no large gap between the two methods. From the results of the five random samples, each of the two emerged victorious and defeated, possibly due to the chance of random sampling. We believe that another important reason is that the *Classification* method, despite the additional support provided by the supervised signals, only undergoes one backward pass, which makes the gradient information generated by each token more contingent, and the model suffers from more noise compared to the *Mean* method.

| Method | Random | Prec. | Rec. | F1 |
|---|---|---|---|---|
| Mean | 0 | 94.58 | 94.63 | 94.60 |
| Classification | 0 | 94.88 | 94.57 | **94.73** |
| Mean | 1 | 90.37 | 87.74 | **89.03** |
| Classification | 1 | 89.63 | 86.29 | 87.93 |
| Mean | 2 | 83.45 | 83.09 | 83.37 |
| Classification | 2 | 85.42 | 83.46 | **84.43** |
| Mean | 3 | 93.55 | 92.89 | **93.22** |
| Classification | 3 | 93.35 | 92.89 | 93.12 |
| Mean | 4 | 96.33 | 96.34 | **96.34** |
| Classification | 4 | 96.18 | 96.20 | 96.19 |
| Mean | average | 91.66 | **90.94** | **91.31** |
| Classification | average | **91.89** | 90.68 | 91.28 |

Table 5: Comparison of different forward-backward procedures.