Relation Repository Based Adaptive Clustering for Open Relation Extraction

Ke Chang and Ping Jian(✉)

Beijing Institute of Technology, Beijing, China
{changke,pjian}@bit.edu.cn

Abstract. Clustering-based relation discovery is one of the important methods in the field of open relation extraction (OpenRE). However, samples residing in semantically overlapping regions often remain indistinguishable. In this work, we propose an adaptive clustering method based on a relation repository to explicitly model the semantic differences between clusters to mitigate the relational semantic overlap in unlabeled data. Specifically, we construct difficult samples and use bidirectional margin loss to constrain the differences of each sample and apply self-supervised contrastive learning to labeled data. Combined with contrastive learning of unlabeled data, we construct a relation repository to explicitly model the semantic differences between clusters. Meanwhile, we place greater emphasis on the difficult samples located on the boundary, enabling the model to adaptively adjust the decision boundary, which lead to generate cluster-friendly relation representations to improve the effect of open relation extraction. Experiments on two public datasets show that our method can effectively improve the performance of open relation extraction.

Keywords: open relation extraction · contrastive learning · adaptive clustering

1 Introduction

The goal of Open Relation Extraction (OpenRE) is to mine structured information from unstructured text without being restricted by the set of predefined relations in the original text. Methods for dealing with open relation extraction can be roughly divided into two categories. One is Open Information Extraction (OpenIE), which extracts relational phrases of different relational types from sentences. However, this approach is limited by the redundancy of different relation phrases. The other category is unsupervised relation discovery, which focuses on unsupervised relation clustering. Furthermore, the self-supervised signal provides an optimization direction for relation clustering. Hu et al. [6] proposed a relation-oriented clustering method to predict both predefined relations and novel relations.

In current methods, the encoder is guided to update relation representations using pseudo-labels generated through clustering. However, these methods still

H. Wang et al. (Eds.): CCKS 2023, CCIS 1923, pp. 95–106, 2023.
https://doi.org/10.1007/978-981-99-7224-1_8

face challenges when dealing with difficult samples that are classified incorrectly due to semantic overlap between clusters. Specifically, instances with highly similar contexts but different relation types tend to lie at the boundary of two clusters in the semantic space. As a result, during training, blurred decision boundaries lead to the generation of incorrect guidance signals, causing these instances to oscillate between the two clusters. This phenomenon significantly impedes the accurate semantic description of relations and the appropriate categorization of relation types.

By integrating the instance and class perspectives, we propose a novel approach that leverages a relational repository to store relation representations in clusters after each epoch. This allows us to address the limitation of optimizing instances and clusters simultaneously under a single perspective. We utilize cluster representations to capture and model the semantic distinctions between clusters, enabling the model to effectively learn and optimize the decision boundary. In addition, the introduction of the sample attention mechanism on the decision boundary during the training process can improve the classification of difficult samples from the perspective of clustering.

The major contributions of our work are as follows: (1) For predefined relations, bidirectional margin loss is used to distinguish difficult samples, and instance-level self-supervised contrastive learning is enhanced for knowledge transfer. (2) For novel relations, cluster semantics are aligned with relational semantics on the basis of constructing a relation repository, and weights are used to emphasize difficult samples in training. (3) Experiment results and analyses on two public datasets demonstrate the effectiveness of our proposed method.

2 Related Work

Open relation extraction is used for extracting new relation types. The Open Information Extraction (OpenIE) regards the relation phrases within the sentence as individual relation types, but the same relation often has multiple surface forms, resulting in redundant relation facts.

Unsupervised relation clustering methods focus on relation types. Recently, Hu et al. [6] is an adaptive clustering model to iteratively get pseudo-labels on the BERT-encoded relation representations, and then used the pseudo-labels as self-supervised signals to train relation classifier and optimize the encoder. Zhao et al. [16] followed SelofORE's iterative generation pseudo-label scheme as part of unsupervised training. In order to obtain the relation information from the predefined data, they learned low-dimensional relation representations oriented to clustering constraints with the help of labeled data. This method does not need to design complex clustering algorithms to complete the identification of relational representations. Different from them, we proposed a method based on relation repository to explicitly model the difference in cluster semantics.

3 Method

The training data set D includes predefined relation data $D^l = \{(\boldsymbol{s}_i^l, y_i^l)\}_{i=1}^N$ and novel relation data set $D^u = \{\boldsymbol{s}_i^u\}_{i=1}^M$, N and M represent the number of relation instances in each data set, $\boldsymbol{s}_i^l$ in D^l and $\boldsymbol{s}_i^u$ in D^u are all relation instances, including the sentence, as well as the head entity and tail entity in the text. And the $y_i^l \in \mathcal{Y}^l = \{1, ..., C^l\}$ is the relation label corresponding to the instance $\boldsymbol{s}_i^l$, the label is visible to the model during training, and the one-hot vector corresponding to y_i^l is represented as $\boldsymbol{y}_i^l$. C^u is provided as prior knowledge to the model.

Our goal is to automatically cluster relation instances in all unlabeled datasets into C^u categories, in particular, $C^l \cap C^u = \emptyset$. Considering that the data to be predicted in real-world scenarios does not only come from unlabeled data, we use labeled and unlabeled data to evaluate the discriminative ability of the model during testing.

3.1 Relation Representations

Given a sentence $\boldsymbol{x} = (x_1, \ldots, x_T)$, where T is the number of tokens in the sentence, e_h and e_t are two entities in the sentence and marked with their start and end positions. The combination of them forms a relation instance $\boldsymbol{s} = (\boldsymbol{x}, e_h, e_t)$.

For the sentence $\boldsymbol{x}$ of the relation instance $\boldsymbol{s}$, each token is encoded as $h \in R^d$ by the encoder $\boldsymbol{f}$, where d represents the output dimension. The $\boldsymbol{f}$ here is the pre-trained language model BERT [2]. We use the maximum pooling of the token hidden layer vectors related to the head entity and the tail entity to obtain the hidden layer vectors of the two entities:

$$\begin{aligned} h_1, \ldots, h_T &= \text{BERT}(x_1, \ldots, x_T) \\ h_{ent} &= \text{MAXPOOL}([h_s, \ldots, h_e]) \end{aligned} \tag{1}$$

where $h_{ent} \in R^d$ represents the entity representation, s and e represent the start and end positions of an entity, respectively. The concatenation of the head entity representation h_{head} and the tail entity representation h_{tail} is regarded as a relation representation, $[,]$ represents the concatenation operation:

$$\boldsymbol{z}_i = [h_{head}, h_{tail}] \tag{2}$$

where the relation representation $\boldsymbol{z}_i \in R^{2\times d}$.

3.2 Bidirectional Margin Loss

To create a sample with the same relation type but different contexts from the original, we randomly substitute the head entity and tail entity with other words of the same entity type, and the representation of new sample is recorded as $\boldsymbol{z}_i^+$. Furthermore, we randomly choose an instance of a different relation type from the original instance and replace its head entity and tail entity with synonyms found in the original instance. This allows us to construct a sample $\boldsymbol{z}_i^-$ with a similar context but a distinct relation type.

In order to measure the difference between two difficult samples in the labeled data in the same semantic space, the loss L^H is used to limit the difference between the cosine similarity between the original sample and the two difficult samples to the range of $[-m_2, -m_1]$:

$$\begin{aligned}\mathcal{L}^H =& max(0, sim(\boldsymbol{z}_i, \boldsymbol{z}_i^-) - sim(\boldsymbol{z}_i, \boldsymbol{z}_i^+) + m_1) \\ &+ max(0, -sim(\boldsymbol{z}_i, \boldsymbol{z}_i^-) + sim(\boldsymbol{z}_i, \boldsymbol{z}_i^+) - m_2)\end{aligned} \tag{3}$$

where $sim(,)$ is calculated by cosine similarity, the negative of m_1 and the negative of $-m_2$ represent the upper and lower bounds of semantic differences, and m_1 is set to 0.1 and m_2 is 0.2 during training.

3.3 Knowledge Transfer

The objective of knowledge transfer is to obtain information pertaining to relation representations from labeled data and learn relation representations that can be used to cluster unknown categories. In this paper, contrastive learning is used for joint training on mixed datasets to transfer relational knowledge from labeled data to unlabeled data. First we use the positive samples in Sect. 3.2 to construct a positive sample set.

In each batch, for relation instance $\boldsymbol{s}_i$ in dataset D, where $i \in \mathcal{N} = \{1, \dots, N\}$ is the sample number in the same batch, after obtaining the relation representation $\boldsymbol{z}_i$ through relation encoding, follow the traditional contrastive learning strategy, using NCE [4] as the contrastive loss function between instances:

$$\mathcal{L}_i^{NCE-I} = -\log \frac{\exp\left(cos(\boldsymbol{z}_i, \hat{\boldsymbol{z}}_i)/\tau\right)}{\sum_n \mathbb{1}_{[n \neq i]} \exp\left(cos(\boldsymbol{z}_i, \hat{\boldsymbol{z}}_n)/\tau\right)} \tag{4}$$

where $\hat{\boldsymbol{z}}_i$ represent a positive example of $\boldsymbol{z}_i$, τ is the temperature coefficient, $\mathbb{1}_{[n \neq i]}$ means that the expression value is 1 if and only if n is not equal to i, otherwise it is 0.

Unlike traditional self-supervised contrastive learning tasks, there are labeled data in each batch, in order to fully learn the relational knowledge of these labeled data, we use an additional loss. Except for the constructed positive samples, all instances consistent with the current instance label are regarded as more positive samples, while other class instances of the same batch are negative samples. Since the instances of the same category are in the same positive sample set, it indirectly constrains the distribution consistency within the class, and the loss function is as below:

$$\mathcal{L}_i^{NCE-L} = -\frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp\left(cos(\boldsymbol{z}_i, \boldsymbol{z}_p)/\tau\right)}{\sum_n \mathbb{1}_{[n \neq i]} \exp\left(cos(\boldsymbol{z}_i, \boldsymbol{z}_n)/\tau\right)} \tag{5}$$

where $P(i) = \{p \in \mathcal{N} \backslash i : y_p = y_i\}$ represents the set of sample numbers with the same label with the ith instance $\boldsymbol{s}_i$ in a batch. For unlabeled datasets, $P(i) = \emptyset$, $\mathcal{L}_i^{NCE-L} = 0$. We construct the contrastive learning loss:

$$\mathcal{L}^{CL} = \frac{1}{N}\sum_{i}^{N}((1-\lambda)\mathcal{L}_i^{NCE-I} + \lambda\mathcal{L}_i^{NCE-L}) \tag{6}$$

where $\mathcal{L}^{NCE-I}$ only has a pair of positive samples, $\mathcal{L}^{NCE-L}$ use samples of the same relational type as the positive sample set, and constrain the encoder to learn representations that are sensitive to the semantic features of relations.λ is used to balance $\mathcal{L}_i^{NCE-I}$ and $\mathcal{L}_i^{NCE-L}$, avoiding the overfitting of predefined relation.

3.4 Adaptive Clustering

Adaptively adjusting the clustering boundary method is used for unlabeled data clustering, after each training epoch, each sample's pseudo-label is modified to the label set $\mathcal{Y} = \{\hat{y}_1, \ldots, \hat{y}_{BN}\}$, $\hat{y}_i \in [1, C^u]$, where B is the batch number of unlabeled data sets.

In order to facilitate the measurement of the association of cross-category instances with different categories, we use a repository set of size $BN/(C^u - 1)$ $\mathcal{M} = \{M_1, \ldots, M_{C^u}\}$ to store the enhanced instance of each category. For the positive sample representation $\hat{\boldsymbol{z}}^u$ with the current pseudo-label $\hat{y}_i$, other positive sample data except $M_{\hat{y}_i}$ are used as comparison sets Q_i, $Q_i = \{\hat{\boldsymbol{z}}^u | \hat{\boldsymbol{z}}^u \in M_j \quad \forall j \in [1, C^u] \quad and \quad j \neq \hat{y}_i\}$. After each backpropagation, the new relation representation $\hat{\boldsymbol{z}}^u$ enters the corresponding queue $M_{\hat{y}_i}$, and the oldest representation added to the queue will be removed. The repository set maintains instances of each category, which can be used as a basis to realize the division of relational types. The process flow of this module for unlabeled data is shown in Fig. 1, each category corresponds to a list to store related instances.

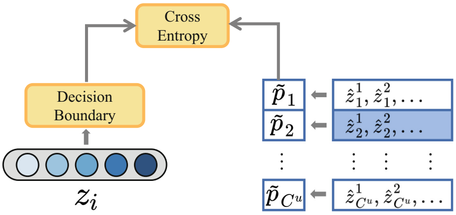

Cross Entropy
Decision Boundary
z_i
$\tilde{p}_1$
$\tilde{p}_2$
$\tilde{p}_{C^u}$
$\hat{z}_1^1, \hat{z}_1^2, \ldots$
$\hat{z}_2^1, \hat{z}_2^2, \ldots$
$\hat{z}_{C^u}^1, \hat{z}_{C^u}^2, \ldots$

Fig. 1. Adaptive Clustering

In order to discover new relations using relation representations, we update decision boundaries by maximizing the intra-cluster similarity and minimizing the inter-cluster similarity and then updating the representations according to the relation repository. The instance representations of each category stored independently are used to construct the cluster center. For the current instance representation $\boldsymbol{z}_i^u$, we use $\tilde{p}_{i,j}$ to calculate the probability that it belongs to the category j:

$$\tilde{p}_{i,j} = \frac{\sum_{\hat{\boldsymbol{z}}^u \in M_j} \exp\left(\cos\left(\boldsymbol{z}_i^u, \hat{\boldsymbol{z}}^u\right)/\tau\right)}{\sum_{j'=1}^{C^u} \sum_{\hat{\boldsymbol{z}}^u \in M_{j'}} \exp\left(\cos\left(\boldsymbol{z}_i^u, \hat{\boldsymbol{z}}^u\right)/\tau\right)} \tag{7}$$

where τ is the temperature coefficient. This formulation measures the semantic similarity of the current instance representation to instances of all categories. The clustering decision boundary is shown below:

$$p_i = \text{Softmax}\left(W^\top \boldsymbol{z}_i^u + b\right) \in \mathcal{R}^{C^u} \tag{8}$$

where W and b are the parameters of the decision boundary, and $\boldsymbol{z}_i^u$ is mapped to a C^u dimensional vector, each dimension represents the probability $p_{i,j}$ of the corresponding category.

To align class semantics with relation categories, we minimize the cross-entropy between the cluster assignment $\tilde{p}_i$ based on the semantic similarity in the feature space and the prediction p_i generated based on the decision boundary:

$$\mathcal{L}^{CD} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{C^u} \tilde{p}_{i,j} \log p_{i,j} \tag{9}$$

Due to the setting of relation repositories, samples are assigned to the most similar category under the constraint of loss, while according to the adaptive decision boundary, relation repositories are updated in time with the semantic features corresponding to them. Following each epoch of training, the parameters of the encoder and the decision boundary are optimized, the label of the instance is updated by maximum likelihood estimation, and the relation repository is updated according to the label:

$$\hat{y}_i = \underset{j}{argmax}\ p_{i,j}, j \in \{1, \ldots, C^u\} \tag{10}$$

During training, some samples may change label repeatedly in adjacent epochs, which is formalized as:

$$\boldsymbol{s}_i^e = \boldsymbol{s}_i^{e-1} + \mathbb{1}[\hat{y}_i^e \neq \hat{y}_i^{e-1}] \tag{11}$$

where $\boldsymbol{s}_i^e$ represents the instance $\boldsymbol{s}_i$ in the eth epoch of training. These samples may be the difficult samples at the decision boundary. With the help of the attention mechanism, higher weights are given to these samples so that the model can achieve the correct prediction of the difficult samples:

$$w_i^e = \frac{\boldsymbol{s}_i^e}{\sum_j^N \boldsymbol{s}_j^e} \tag{12}$$

where w_i^e represents the weight of $\boldsymbol{z}_i^u$ in the eth epoch of training.

We can update the weights in the instance discriminative loss $\mathcal{L}^{NCE-I}$, and update $\mathcal{L}^{CL}$:

$$\mathcal{L}^{NCE-I} = \sum_{i=1}^{N} w_i^e \mathcal{L}_i^{NCE-I} \tag{13}$$

$$\mathcal{L}^{CL} = (1-\lambda)\mathcal{L}^{NCE-I} + \frac{\lambda}{N}\sum_{i}^{N}(\mathcal{L}_i^{NCE-L}) \tag{14}$$

We set a cross-entropy loss in order to avoid the catastrophic forgetting phenomenon of predefined relations in the process of guiding the discovery of new relations. We use the softmax layer σ to map the relation representation $\boldsymbol{z}_i^l \in \mathbb{R}^{C^l}$ to a posterior distribution $p_c = \sigma(\boldsymbol{z}_i^l)$ with dimension C^l. The loss function is defined as follows:

$$\mathcal{L}^{CE} = -\sum_{c=1}^{C_l} y_c \log(p_c) \tag{15}$$

The total loss is:

$$\mathcal{L} = \alpha\mathcal{L}^{H} + \mathcal{L}^{CL} + \mathcal{L}^{CD} + \beta\mathcal{L}^{CE} \tag{16}$$

where α and β are hyperparameters used to balance the overall loss.

4 Experiments

4.1 Datasets

To assess the performance of our method, we conduct experiments on two relation extraction datasets. **FewRel** [5] consists of texts from Wikipedia that are automatically annotated with Wikidata triple alignments in a far-supervised manner followed by manual inspection. It contains 80 relation types, there are 700 instances in each type. **TACRED** [15] is a large-scale human-annotated relation extraction dataset, including 41 relation types.

For FewRel, 64 types of relation in the original training set will be used as labeled data, and the 16 types of relation in the original verification set will be used as unlabeled data sets to discover new relations. Each type of data is divided into the training set and the test set according to 9:1. For TACRED, after removing instances labeled “No Relation”, the remaining 21,773 instances are used for training and evaluation. Afterward, the 0–30 relation types are regarded as labeled datasets, and the 31–40 relation types are regarded as unlabeled datasets.In each dataset, 1/7 of the data is randomly selected as the test set, and the rest of the data is divided into the train set.

We use B^3 [1], $V-measure$ [11] and ARI [7] to evaluate the performance of the model, they are used to measure the accuracy and recall of clustering, the uniformity and completeness of clusters, and the consistency between clusters and the true distribution.

4.2 Baselines

We select these OpenRE baselines for comparison:

Discrete-state Variational Autoencoder (VAE) [10]. VAE exploits the reconstruction of entities and predicted relations to achieve open-domain relation extraction.

HAC with Re-weighted Word Embeddings (RW-HAC) [3]. RW-HAC utilizes entity type and word embedding weights as relational features for clustering.

Entity Based URE (Etype+) [12]. Etype+ relies on entity types and uses a link predictor and two additional regularizers on top of VAE.

Relational Siamese Network (RSN) [13]. RSN learns the similarity of predefined relation representations from labeled data and transfers relation knowledge to unlabeled data to identify new relations.

RSN with BERT Embedding (RSN-BERT) [13]. This method is based on the RSN model and uses word embeddings encoded by BERT instead of standard word vectors.

Self-supervised Feature Learning for OpenRE (SelfORE) [6]. SelfORE uses a large-scale pre-trained language model and self-supervised signals to achieve adaptive clustering of contextual features.

Relation-Oriented Open Relation Extraction (RoCORE) [16]. RoCORE learns relation-oriented representations from labeled data with predefined relations and uses iterative joint training to reduce the bias caused by labeled data.

The unsupervised benchmark models include VAE, RE-HAC, EType+, the self-supervised benchmark model is SelfORE, and the supervised benchmark models include RSN, RSN-BERT, and RoCORE.

4.3 Implementation Details

Referring to the settings of the baseline model, we use BERT-Base-uncased to initialize the word embedding. At the same time, in order to avoid overfitting, we refer to the settings of Zhao et al. [16] and only fine-tune the parameters of Layer 8. We use Adam [8] as the optimizer, 5e−4 as learning rate, and the batch size is 100. α is 5e−4, 1e−3 on the FewRel and TACRED, β is set to 0.8, λ is set to 0.35 on the two datasets, this parameter depends on the importance of hard samples in predefined relations on different datasets. We use the "merge and split" method [14] when updating pseudo-labels to avoid cluster degradation caused by unbalanced label distribution. All experiments are trained on GeForce RTX A6000 with 48 GB memory.

4.4 Main Results

The main results are shown in Table 1. The method proposed in this paper exceeds the strong baseline model RoCORE on three main evaluation indicators

Table 1. Experimental results produced by baselines and proposed model on FewRel and TACRED in terms of B^3, V-measure, ARI. The horizontal line divides unsupervised and supervised methods.

Dataset	Method	B^3			$V - measure$			ARI
		Prec.	Rec.	F_1	Hom.	Comp.	F_1	
FewRel	VAE	30.9	44.6	36.5	44.8	50.0	47.3	29.1
	RW-HAC	25.6	49.2	33.7	39.1	48.5	43.3	25.0
	EType+	23.8	48.5	31.9	36.4	46.3	40.8	24.9
	SelfORE	67.2	68.5	67.8	77.9	78.8	78.3	64.7
	RSN	48.6	74.2	58.9	64.4	78.7	70.8	45.3
	RSN-BERT	58.5	**89.9**	70.9	69.6	**88.9**	78.1	53.2
	RoCORE	**75.2**	84.6	79.6	83.8	88.3	86.0	70.9
	Ours	78.5	82.6	**80.5**	85.6	88.7	**87.1**	**72.4**
TACRED	VAE	24.7	56.4	34.3	20.8	36.2	26.4	15.9
	RW-HAC	42.6	63.3	50.9	46.9	59.7	52.6	28.1
	EType+	30.2	80.3	43.9	26.0	60.7	36.4	14.3
	SelfORE	57.6	51.0	54.1	63.0	60.8	61.9	44.7
	RSN	62.8	63.4	63.1	62.4	66.3	64.3	45.9
	RSN-BERT	79.5	**87.8**	83.4	84.9	87	85.9	75.6
	RoCORE	87.1	84.9	86.0	**89.5**	88.1	88.8	82.1
	Ours	**85.9**	87.3	**86.6**	89.1	**89.3**	**89.2**	**82.6**

B^3F_1, $V-measureF_1$ and ARI on all datasets, bringing 0.9%/0.6%, 1.1%/0.4% and 1.5%/0.5% growth respectively. Utilizing RoCORE and conducting paired t-tests on key performance indicators through multiple experiments, the one-tailed p-values on the two datasets are as follows: 0.002/0.024, 0.011/0.019, and 0.004/0.005, all of which are less than 0.05 indicates that our method exhibits significant differences from the RoCORE method in terms of the aforementioned indicators. It reveals that the method in this paper can effectively use the relation repository sets to model the semantic differences of different relations compared with other models. The encoder is then encouraged to generate cluster-oriented deep relation representations.

4.5 Ablation Analysis

In order to deeply analyze the influence of each key module on the performance of the model, we construct some ablation experiments, and the experiment results are the average results of multiple experiments (Table 2).

Bidirectional Margin Loss. Bidirectional margin loss can handle difficult samples better. Comparative analysis reveals that the model's performance on both datasets deteriorates after removing the margin loss, with a more pronounced decline observed in TACRED. This suggests that difficult samples within predefined relations have varying effects on different datasets.

Knowledge Transfer. Knowledge transfer of predefined relations greatly facilitates the discovery of new relations. Notably, the impact of knowledge transfer

Table 2. Abalation study of our method.

Dataset	Method	B^3			$V-measure$			ARI
		Prec.	Rec.	F_1	Hom.	Comp.	F_1	
FewRel	**Ours**	78.5	82.6	**80.5**	85.6	88.7	**87.1**	**72.4**
	w/o margin loss	78.3	82.4	80.3	85.5	88.1	86.8	72.2
	w/o knowledge transfer	77.1	73.8	75.4	83.3	84.7	84.0	68.7
	w/o ID training	74.6	76.6	75.6	81.6	85.3	83.4	69.8
	w/o weight w_i^e	74.6	82.4	78.3	82.3	87.2	84.7	69.3
TACRED	**Ours**	85.9	87.3	**86.6**	89.1	89.3	**89.2**	**82.6**
	w/o margin loss	86.4	86.0	86.2	89.2	88.6	88.9	82.1
	w/o knowledge transfer	83.9	84.7	84.3	87.2	87.0	87.1	79.1
	w/o ID training	85.3	79.5	82.3	85.6	87.0	86.3	78.2
	w/o weight w_i^e	85.6	81.9	83.7	88.9	86.1	87.5	78.6

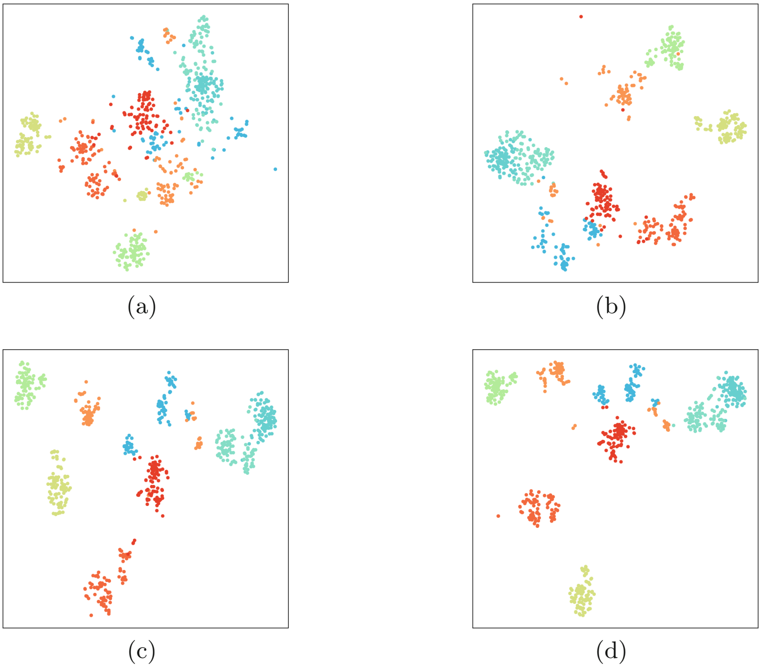
on the FewRel dataset, in the absence of supervised contrastive loss for predefined relations, is more substantial than on TACRED. This underscores the beneficial role of knowledge transfer in enabling the encoder to learn relation representations.

Adaptive Clustering. Adaptive clustering holds equal importance in conjunction with knowledge transfer of predefined relations. Despite employing the knowledge within the relation repository to update pseudo-labels as a substitute, its effectiveness remains inferior to the cluster assignment guided by the clustering boundary. This highlights the efficacy of iteratively updating the decision boundary for the clustering of new relations.

Sample Attention Mechanism. Incorporating the difficult sample attention mechanism enhances the model's ability to discriminate between classes. The removal of the weighting strategy significantly diminishes the clustering effect on different datasets, underscoring the importance of emphasizing difficult samples with ambiguous semantics to improve the model's class discrimination ability.

4.6 Visualization Analysis

In order to show intuitively how our method helps refine the relation representation space, t-SNE [9] is used to visualize each relation representation in the semantic space. We randomly select 8 categories from the training set of FewRel, with a total of 800 relation representations, and reduce the dimension of each representation from 2×768 to 2 dimensions. The change of the relational semantic space during the training process is shown in Fig. 2, after training for 10, 30, and 52 epochs, the representation in the cluster is more compact than before, and the boundary between each cluster is more clear, and the clusters of each relation category have been aligned with the semantics.


(a)

(b)

(c)

(d)

Fig. 2. Visualization of the relation representations

5 Conclusion

In this paper, we propose a relation repository-based adaptive clustering for open relation extraction. Our main contribution is to enhance the model's capability to classify difficult samples. The proposed method leverages bidirectional margin loss and adaptive clustering to enhance the prediction performance for both predefined and novel relations. Experiments and analysis demonstrate the effectiveness of our method.

Acknowledgements. We thank the anonymous reviewers. This work is supported by the Open Project Program of the National Defense Key Laboratory of Electronic Information Equipment System Research under Grant DXZT-JCZZ-2017-009.

References

1. Bagga, A., Baldwin, B.: Entity-based cross-document coreferencing using the vector space model. In: Proceeding of ACL, pp. 79–85 (1998)
2. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL, pp. 4171–4186 (2019)
3. ElSahar, H., Demidova, E., Gottschalk, S., Gravier, C., Laforest, F.: Unsupervised open relation extraction. CoRR abs/1801.07174 (2018)

4. Gutmann, M., Hyvärinen, A.: Noise-contrastive estimation: a new estimation principle for unnormalized statistical models. J. Mach. Learn. Res. **9**, 297–304 (2010)
5. Han, X., et al.: Fewrel: a large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In: Proceedings of EMNLP, pp. 4803–4809 (2018)
6. Hu, X., Wen, L., Xu, Y., Zhang, C., Yu, P.S.: Selfore: self-supervised relational feature learning for open relation extraction. In: Proceedings of EMNLP, pp. 3673–3682 (2020)
7. Hubert, L.J., Arabie, P.: Comparing partitions. J. Classif. **2**, 193–218 (1985)
8. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015, Conference Track Proceedings (2015)
9. Laurens, V.D.M., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(2605), 2579–2605 (2008)
10. Marcheggiani, D., Titov, I.: Discrete-state variational autoencoders for joint discovery and factorization of relations. Trans. Assoc. Comput. Linguist. **4**(2), 231–244 (2016)
11. Rosenberg, A., Hirschberg, J.: V-measure: a conditional entropy-based external cluster evaluation measure. In: Proceedings of EMNLP, pp. 410–420 (2007)
12. Tran, T.T., Le, P., Ananiadou, S.: Revisiting unsupervised relation extraction. In: Proceedings of ACL, pp. 7498–7505 (2020)
13. Wu, R., et al.: Open relation extraction: Relational knowledge transfer from supervised data to unsupervised data. In: Proceedings of EMNLP, pp. 219–228 (2019)
14. Zhan, X., Xie, J., Liu, Z., Ong, Y., Loy, C.C.: Online deep clustering for unsupervised representation learning. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6687–6696 (2020)
15. Zhang, Y., Zhong, V., Chen, D., Angeli, G., Manning, C.D.: Position-aware attention and supervised data improve slot filling. In: Proceedings of EMNLP, pp. 35–45 (2017)
16. Zhao, J., Gui, T., Zhang, Q., Zhou, Y.: A relation-oriented clustering method for open relation extraction. In: Proceedings of EMNLP, pp. 9707–9718 (2021)