A Relation-Oriented Clustering Method for Open Relation Extraction

Jun Zhao[1], **Tao Gui**[2*], **Qi Zhang**[1*], **Yaqian Zhou**[1]

[1]School of Computer Science, Shanghai Key Laboratory of Intelligent Information Processing,
Fudan University, Shanghai, China
[2]Institute of Modern Languages and Linguistics, Fudan University
{zhaoj19,tgui,qz,zhouyaqian}@fudan.edu.cn

Abstract

The clustering-based unsupervised relation discovery method has gradually become one of the important methods of open relation extraction (OpenRE). However, high-dimensional vectors can encode complex linguistic information which leads to the problem that the derived clusters cannot explicitly align with the relational semantic classes. In this work, we propose a relation-oriented clustering model and use it to identify the novel relations in the unlabeled data. Specifically, to enable the model to learn to cluster relational data, our method leverages the readily available labeled data of pre-defined relations to learn a relation-oriented representation. We minimize distance between the instance with same relation by gathering the instances towards their corresponding relation centroids to form a cluster structure, so that the learned representation is cluster-friendly. To reduce the clustering bias on predefined classes, we optimize the model by minimizing a joint objective on both labeled and unlabeled data. Experimental results show that our method reduces the error rate by 29.2% and 15.7%, on two datasets respectively, compared with current SOTA methods.

1 Introduction

Relation extraction (RE), a crucial basic task in the field of information extraction, is of the utmost practical interest to various fields including web search (Xiong et al., 2017), knowledge base completion (Bordes et al., 2013), and question answering (Yu et al., 2017). However, conventional RE paradigms such as supervision and distant supervision are generally designed for pre-defined relations, which cannot deal with new emerging relations in the real world.

Under this background, open relation extraction (OpenRE) has been widely studied for its use
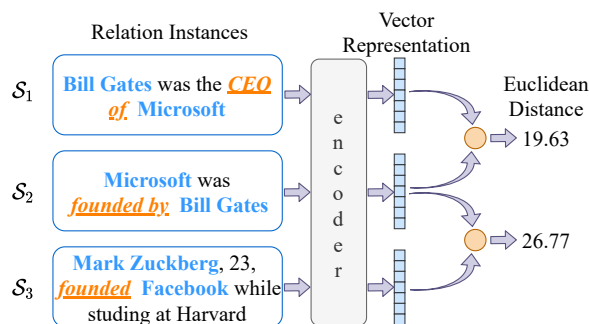
---
*Corresponding authors.

Figure 1: Although both instances $\mathcal{S}_2$ and $\mathcal{S}_3$ express *founded* relation while $\mathcal{S}_1$ expresses *CEO* relation, the distance between $\mathcal{S}_1$ and $\mathcal{S}_2$ is still smaller than that between $\mathcal{S}_2$ and $\mathcal{S}_3$. This is because there may be more similar surface information (e.g. word overlapping) or syntactic structure between $\mathcal{S}_1$ and $\mathcal{S}_2$, thus the derived clusters cannot explicitly align with relations.

in extracting new emerging relational types from open-domain corpora. The approaches used to handle open relations roughly fall into one of two groups. The first group is open information extraction (OpenIE) (Etzioni et al., 2008; Yates et al., 2007; Fader et al., 2011), which directly extracts related phrases as representations of different relational types. However, if not properly canonicalized, the extracted relational facts can be redundant and ambiguous. The second group is unsupervised relation discovery (Yao et al., 2011; Shinyama and Sekine, 2006; Simon et al., 2019). In this type of research, much attention has been focused on unsupervised clustering-based RE methods, which cluster and recognize relations from high-dimensional representations (Elsahar et al., 2017). Recently, the self-supervised signals in pretrained language model are further exploited for clustering optimization (Hu et al., 2020).

However, many studies show that high-dimensional embeddings can encode complex linguistic information such as morphological (Peters et al., 2018), local syntactic (Hewitt and Manning, 2019), and longer range semantic

information (Jawahar et al., 2019). Consequently, the distance of representation is not completely consistent with relational semantic similarity. Although Hu et al. (2020) use self-supervised signals to optimize clustering, there is still no guarantee that the learned clusters will explicitly align with the desired relational semantic classes (Xing et al., 2002). As shown in Figure 1, we use the method proposed by Hu et al. (2020) to get the instance representations. Although both instances $\mathcal{S}_2$ and $\mathcal{S}_3$ express the *founded* relation, the euclidean distance between them is larger than that between $\mathcal{S}_1$ and $\mathcal{S}_2$, which express different relation. Obviously, the clustering algorithm tends to group instances $\mathcal{S}_1$ and $\mathcal{S}_2$ together, rather than $\mathcal{S}_2$ and $\mathcal{S}_3$ which express the same relation.

In this work, we propose a relation-oriented clustering method. To enable the model to learn to cluster relational data, pre-defined relations and their existing labeled instances are leveraged to optimize a non-linear mapping, which transforms high-dimensional entity pair representations into relation-oriented representations. Specifically, we minimize distance between the instances with same relation by gathering the instances representation towards their corresponding relation centroids to form the cluster structure, so that the learned representation is cluster-friendly. In order to reduce the clustering bias on the predefined classes, we iteratively train the entity pair representations by optimizing a joint objective function on the labeled and unlabeled subsets of the data, improving both the supervised classification of the labeled data, and the clustering of the unlabeled data. In addition, the proposed method can be easily extended to incremental learning by classifying the pre-defined and novel relations with a unified classifier, which is often desirable in real-world applications. Our experimental results show that our method outperforms current state-of-the-art methods for OpenRE. Our codes are publicly available at Github[*].

To summarize, the main contributions of our work are as follows: (1) we propose a novel relation-oriented clustering method RoCORE to enable model to learn to cluster relational data; (2) the proposed method achieves the incremental learning of unlabeled novel relations, which is often desirable in real-world applications; (3) experimental results show that our method reduces

the error rate by 29.2% and 15.7%, on two real-world datasets respectively, compared with current state-of-the-art OpenRE methods.

2 Related Work

Open Relation Extraction. To meet the needs of extracting new emerging relation types, many efforts have been undertaken to exploring methods for open relation extraction (OpenRE). The first line of research is Open Information Extraction (Etzioni et al., 2008; Yates et al., 2007; Fader et al., 2011), in which relation phrases are extracted directly to represent different relation types. However, using surface forms to represent relations results in an associated lack of generality since many surface forms can express the same relation. Recently, unsupervised clustering-based RE methods is attracting lots of attentions. Elsahar et al. (2017) proposed to extract and cluster open relations by re-weighting word embeddings and using the types of named entities as additional features. Hu et al. (2020) proposed to exploit weak, self-supervised signals in pretrained language model for adaptive clustering on contextualized relational features. However, the self-supervised signals are sensitive to the initial representation (Gansbeke et al., 2020) and there is still no guarantee that the learned clusters will align with the relational semantic classes (Xing et al., 2002). Wu et al. (2019) proposed the relation similarity metrics from labeled data, and then transfers the relational knowledge to identify novel relations in unlabeled data. Different from them, we propose a relation-oriented method explicitly clustering data based on relational information.

Knowledge in High-Dimensional Vector. Pre-trained static and contextual word representations can provide valuable prior knowledge for constructing relational representations (Soares et al., 2019; Elsahar et al., 2017). Peters et al. (2018) showed that different neural architectures (e.g., LSTM, CNN, and Transformers) can hierarchically structure linguistic information that varies with network depth. Recently, many studies (Jawahar et al., 2019; Clark et al., 2019; Goldberg, 2019) have shown that such hierarchy also exists in pre-training models like BERT. These results suggest that high-dimensional embeddings, independent of model architecture, learn much about the structure of language. Directly clustering on these high-dimensional embeddings should hardly produce

---
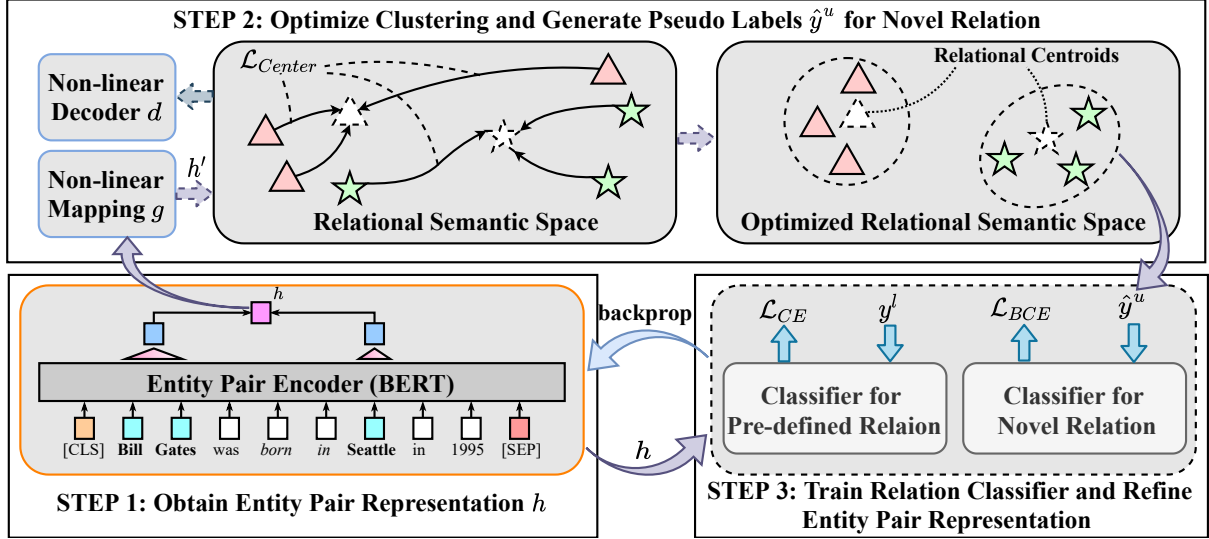
[*]https://github.com/Ac-Zyx/RoCORE.

Figure 2: Overview of our RoCORE method. At the first step, we encode both the labeled and unlabeled instances in to entity pair representations. Then the entity pair representations are transformed to relation-oriented representations by gathering towards their relational centroids in the second step. Finally, based on the pseudo labels generated by clustering on unlabeled data, we optimize the entity pair representations and classifier by minimizing a joint objective function to reduce the clustering bias on predefined classes. The above three steps are performed iteratively to gradually improve model performance on novel relations.

ideal clusters in our desired way, which motivates us to extend current unsupervised clustering-based RE methods to learn the representations tailored for clustering relational data.

3 Approach

In this work, we propose a relation-oriented clustering method, which takes advantage of the relational information in the existing labeled data to enable model to learn to cluster relational data. In order to reduce the clustering bias on the predefined classes, we iteratively train the entity pair representations by optimizing a joint objective function on the labeled and unlabeled subsets of the data, improving both the supervised classification of the labeled data, and the clustering of the unlabeled data. The proposed method is shown in Figure 2.

Specifically, given an unlabeled dataset $\mathcal{D}^u = \{s_i^u\}_{i=1,...,M}$ of relational instances s_i^u, our goal is to automatically cluster the relational instances into a number of classes C^u, which we assume to be known a priori. To enable the model to learn to cluster data, we incorporate a second labeled dataset of pre-defined relations $\mathcal{D}^l = \{(s_i^\ell, y_i^\ell)\}_{i=1,...,N}$ where $y_i^\ell \in \{1,...,C^\ell\}$ is the relational label for instance s_i^ℓ.

3.1 Method Overview

We approach the problem by learning a relation-oriented representation, from which the derived clusters can be explicitly aligned with the desired relational semantic classes. As illustrated in Figure 2, we learn the representation and optimize the model by performing three iterative steps:

(1) First, we encode relation instances in $\mathcal{D}^\ell$ and $\mathcal{D}^u$ using the entity pair encoder implemented as the pretrained BERT (Devlin et al., 2018), which takes relation instances $\{s_i^\ell\}_{i=1,...,N}$, and $\{s_j^u\}_{j=1,...,M}$, as input, and output relation representation $\boldsymbol{h}_i^\ell$, $\boldsymbol{h}_j^u$. However, high-dimensional $\boldsymbol{h}$ can encode a mixture of various aspects of linguistic features and the derived clusters from $\boldsymbol{h}$ cannot explicit align with desired relational classes. (2) To make the distance between the representations accurately reflect the relational semantic similarity, the obtained $\boldsymbol{h}_i^\ell$ are transformed to low-dimensional relation-oriented representations $\boldsymbol{h}_i^{\ell\prime}$ by a non-linear mapping $\boldsymbol{g}$. Under the supervision of labels y_i^ℓ in $\mathcal{D}^\ell$, $\boldsymbol{g}$ is optimized by the gathering of $\boldsymbol{h}_i^{\ell\prime}$ towards their relational centroids to form a cluster structure, thereby we obtain $\boldsymbol{h}_j^{u\prime}$ from unlabeled data using the optimized $\boldsymbol{g}$ and generate the pseudo labels $\hat{y}^u$ according to clustering on $\boldsymbol{h}_i^{u\prime}$. (3) Because using labeled data to guide the $\boldsymbol{h}'$ towards their relational centroids will produce

9709

clustering bias on pre-defined relations, it is difficult to directly generate high-quality pseudo labels. To reduce the negative effect of errors in pseudo labels, we optimize classifier and entity pair representations by minimizing a joint objective function, containing terms for both pre-defined and novel relations, using respectively the given labels y^ℓ and generated pseudo label $\hat{y}^u$. Based on the refined entity pair representation h which encode more contextual relational information, the above three steps are performed iteratively to gradually improve the quality of pseudo labels $\hat{y}^u$ and model performance.

3.2 Entity Pair Encoder

Given a relation instance $s_i = (\boldsymbol{x}_i, h_i, t_i)$, which consists of a sentence $\boldsymbol{x}_i = \{x_1, x_2, ..., x_n\}$ and two entity spans $h_i = (s_h, e_h)$, $t_i = (s_t, e_t)$ marking the position of the entity pair, the entity pair encoder $\boldsymbol{f}$ aims to map relation instance s_i to a fixed-length embedding $\boldsymbol{h}_i = \boldsymbol{f}(s_i) \in \mathbb{R}^d$ that encode contextual information in s_i. We adopt BERT (Devlin et al., 2018) as the implemention of our encoder $\boldsymbol{f}$ due to its strong performance on extracting contextual information. Formally:

$$\boldsymbol{h}_1^r, ..., \boldsymbol{h}_n^r = \text{BERT}^r(x_1, ..., x_n) \quad (1)$$

$$\boldsymbol{h}_{ent} = \text{MAXPOOL}(\boldsymbol{h}_s^r, ..., \boldsymbol{h}_e^r) \quad (2)$$

$$\boldsymbol{h}_i = \boldsymbol{h}_{head} \oplus \boldsymbol{h}_{tail}, \quad (3)$$

where r is a hyperparameter that denotes the output layer of BERT. s and e represent start and end position of the corresponding entity respectively. $\oplus$ denotes the concatenation operator. This structure of entity pair representation encoder has been widely used in previous RE methods (Wang et al., 2021; Hu et al., 2020).

3.3 Relation-Oriented Clustering Module

In order to make the distance between representation accurately reflect the relational semantic similarity, the obtained $\{\boldsymbol{h}_i^\ell\}_{i=1,...,N}$ are transformed to low-dimensional relation-oriented representations $\boldsymbol{h}_i^{\ell\prime}$ by a non-linear mapping $\boldsymbol{g}(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^m$. Under the supervision of labels y_i^ℓ in $\mathcal{D}^\ell$, $\boldsymbol{g}$ is optimized by the gathering of $\boldsymbol{h}_i^{\ell\prime}$ towards their relational centroids as follows:

$$\mathcal{L}_{center} = \frac{1}{2N} \sum_{i=1}^N \left\| \boldsymbol{h}_i^{\ell\prime} - \boldsymbol{c}_{y_i} \right\|_2^2 \quad (4)$$

$$\boldsymbol{c}_r = \frac{1}{|\mathcal{D}_r|} \sum_{i \in \mathcal{D}_r} \boldsymbol{h}_i^{\ell\prime}, \quad (5)$$

where $\boldsymbol{c}_r$ denotes the centroids of relation r. The center loss $\mathcal{L}_{center}$ seems reasonable, but problematic. A global optimal solution to minimize $\mathcal{L}_{center}$ is $g(\boldsymbol{h}_i) = \boldsymbol{0}$, which is far from being desired. This motivates us to incorporate a reconstruction term to prevent the semantic space from collapsing. Specifically,, a decoding network $\boldsymbol{d}(\cdot)$ is used to map the representation $\boldsymbol{h}_i'$ back to the original representation $\boldsymbol{h}_i$. Thus, we can derive the following loss function:

$$\mathcal{L}_C = \frac{1}{2N} \sum_{i=1}^N \ell(\boldsymbol{d}(\boldsymbol{h}_i'), \boldsymbol{h}_i) + \lambda \mathcal{L}_{center}, \quad (6)$$

where both the encoder $g(\boldsymbol{h}_i)$ and decoder $\boldsymbol{d}(\boldsymbol{h}_i')$ are implemented as DNN. The function $\ell(\cdot, \cdot) : \mathbb{R}^d \rightarrow \mathbb{R}$ is the least-squares loss $\ell(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|_2^2$ that measures the reconstruction error and other choices such as ℓ_1-norm also can be considered. λ is a hyper-parameter that balances the reconstruction error versus center loss.

Finally, we obtain $\{\boldsymbol{h}_j^{u\prime}\}_{j=1,...,M}$ using the optimized $\boldsymbol{g}$ and generate pseudo labels $\hat{y}^u$ using k-means algorithm as follows:

$$\hat{y}^u = \text{k-means}(\boldsymbol{h}^{u\prime}), \quad (7)$$

3.4 Relation Classification Module

Based on the pseudo labels $\hat{y}^u$ generated by clustering, we can train the classifier and refine entity pair representation $\boldsymbol{h}$ to encode more contextual relation information. Since it's difficult to keep the order of clusters consistent in multiple clustering, instead of using standard cross entropy loss, we propose to use the pairwise similarities for novel relation learning.

$$q_{ij} = \mathbb{1}\{\hat{y}_i^u = \hat{y}_j^u\}, \quad (8)$$

where the symbol q_{ij} denotes whether s_i^u and s_j^u belong to the same cluster. If a pair is from the same cluster, the classifier $\boldsymbol{\eta}^u : \mathbb{R}^d \rightarrow \mathbb{R}^{C^u}$ outputs similar distributions, and vice-versa. Specifically, we use the pair-wise KL-divergence to evaluate the distance of two relation instances. Given a pair of instance s_i^u, s_j^u, their corresponding output distributions are defined as $\mathcal{P} = \boldsymbol{\eta}^u(\boldsymbol{f}(s_i^u))$ and $\mathcal{Q} = \boldsymbol{\eta}^u(\boldsymbol{f}(s_j^u))$. For the pair from the same cluster, the cost is described as:

$$\mathcal{L}^+(s_i^u, s_j^u) = \mathcal{D}_{KL}(\mathcal{P}^*||\mathcal{Q}) + \mathcal{D}_{KL}(\mathcal{Q}^*||\mathcal{P}) \quad (9)$$

$$\mathcal{D}_{KL}(\mathcal{P}^*||\mathcal{Q}) = \sum_{c=1}^{C^u} p_c \log \frac{p_c}{q_c}, \quad (10)$$

where $\mathcal{P}^*$ denotes that $\mathcal{P}$ is assumed to be a constant and each KL-divergence factor $\mathcal{D}_{KL}(\mathcal{P}||\mathcal{Q})$ is a unary function whose gradient is simply $\partial \mathcal{D}_{KL}(\mathcal{P}^*||\mathcal{Q})/\partial \mathcal{Q}$.

If s_i^u, s_j^u comes from different clusters, their output distributions are expected to be different, which can be defined as a hinge-loss function:

$$\mathcal{L}^-(s_i^u, s_j^u) = L_h(\mathcal{D}_{KL}(\mathcal{P}^*||\mathcal{Q}), \sigma) + \\ L_h(\mathcal{D}_{KL}(\mathcal{Q}^*||\mathcal{P}), \sigma) \quad (11)$$

$$L_h(e, \sigma) = \max(0, \sigma - e), \quad (12)$$

and the total loss can be defined as a contrastive loss:

$$\mathcal{L}_{BCE}(s_i^u, s_j^u) = q_{i,j}\mathcal{L}^+(s_i^u, s_j^u) + \\ (1 - q_{ij})\mathcal{L}^-(s_i^u, s_j^u). \quad (13)$$

Note that $\mathcal{L}_{BCE}$ is a symmetric loss w.r.t. s_i^u, s_j^u since $\mathcal{P}$ and $\mathcal{Q}$ are alternatively assumed to be constant in $\mathcal{L}^+$ and $\mathcal{L}^-$. Finally, we get the prediction for a relation instance s_i^u as follows:

$$\hat{y}_i^u = \arg\max_y [\boldsymbol{\eta}^u(\boldsymbol{f}(s_i^u))]_y \quad (14)$$

3.5 Training Methods

3.5.1 Iterative Joint Training

Because using labeled data to guide the $\boldsymbol{h}$ towards their relational centroids will produce clustering bias on pre-defined relations, it is difficult to directly generate high-quality pseudo labels $\hat{y}^u$ for novel relations. To reduce the negative effect of errors in pseudo labels, we incorporate a classifier $\boldsymbol{\eta}^\ell : \mathcal{R}^d \to \mathcal{R}^{C^l}$ for pre-defined relations and refine $\boldsymbol{h}$ by minimizing a joint objective function, containing terms for both pre-defined and novel relations, using respectively the given labels y^ℓ and generated pseudo label $\hat{y}^u$ as follows:

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N} log\boldsymbol{\eta}_{y_i}^\ell(\boldsymbol{h_i}) \quad (15)$$

$$\mathcal{L}_{CLS} = \mathcal{L}_{CE} + \mathcal{L}_{BCE}. \quad (16)$$

The refined entity pair representation $\boldsymbol{h}$ encode more contextual relation information, which in turn promote clustering optimization and generate pseudo labels $\hat{y}^u$ with higher accuracy. We refine representation $\boldsymbol{h}$ and optimize clustering in a iterative manner to gradually improve the quality of the pseudo labels and model performance. This iterative procedure is detailed in Algorithm 1.

Algorithm 1: The RoCORE Method

Input: novel relation dataset $\mathcal{D}^u = \{s_j^u\}$, predefined relation dataset $\mathcal{D}^\ell = \{(s_i^\ell, y_i^\ell)\}$, model parameters Θ, Φ, Ψ for Entity pair encoder, Relation-oriented clustering module, Relation classifiers, respectively, and learning rate η.

1 **for** $epoch \leftarrow 1$ **to** L **do**
2   Pre-train clustering network by minimize reconstruction loss $\Phi = \Phi - \eta\nabla_\Phi\ell(\boldsymbol{g}(\boldsymbol{h}_i'), \boldsymbol{h}_i)$;
3 **end**
4 **repeat**
5   generate pseudo labels $\hat{y}$ by equation 7;
6   refine entity pair representation
7    $\Theta = \Theta - \eta\nabla_\Theta\mathcal{L}_{CLS}$;
8    $\Psi = \Psi - \eta\nabla_\Psi\mathcal{L}_{CLS}$;
9   optimize clustering $\Phi = \Phi - \eta\nabla_\Phi\mathcal{L}_C$;
10 **until** *convergence*;

3.5.2 Incremental Learning Scheme

In real-world settings, when facing a new sentence, we often don't know whether it belongs to pre-defined relations or novel relations. In this work, we explore the incremental learning of novel relations to enable $\boldsymbol{\eta}^l$ to discriminate both pre-defined and novel relations. Under incremental learning settings, we extend the classifier $\boldsymbol{\eta}^l$ to C^u novel relation types, so that $\boldsymbol{\eta}^l : \mathcal{R}^d \to \mathcal{R}^{C^l+C^u}$. Then, the model is trained using cross-entropy loss instead of equation 15 as follows:

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N} log\boldsymbol{\eta}_{y_i}^\ell(\boldsymbol{h}_i) \\ -\frac{\mu(t)}{M}\sum_{j=1}^{M} log\boldsymbol{\eta}_{\hat{y}_j}^\ell(\boldsymbol{h}_j), \quad (17)$$

where we obtain $\hat{y}_j$ using equation 14 and the coefficient $\mu(t)$ balances the cross entropy loss of pre-defined and novel relations. We implemented it as a ramp-up function $\mu(t) = \mu_0 e^{-5(1-\frac{t}{T})^2}$ where t is current epoch and T is the ramp-up length and coefficient $\mu_0 \in \mathbb{R}^+$.

4 Experimental Setup

In this section, we describe the datasets for training and evaluating the proposed method. We also detail the baseline models for comparison.

Finally, we clarify the implementation details and hyperparameter configuration of our method.

4.1 Datasets

We conduct experiments on two relation extraction datasets.

FewRel. Few-Shot Relation Classification Dataset (Han et al., 2018). FewRel is a human-annotated dataset containing 80 types of relations, each with 700 instances. We follow the setting in (Wu et al., 2019) to use the original train set of FewRel, which contains 64 relations, as labeled set with predefined relations, and the original validation set of FewRel, which contains 16 new relations, as the unlabeled set with novel relations to extract. 1,600 instances were randomly selected from the unlabeled set as the test set. The rest of labeled and unlabeled instances are considered as the train set.

TACRED. The TAC Relation Extraction Dataset (Zhang et al., 2017). TACRED is a human-annotated large-scale relation extraction dataset that covers 41 relation types. We remove the instances labeled as `no_relation` and use the remaining 21,773 instances for training and evaluation. Similar to the setting of FewRel, we select the 0-30 relation types as labeled set with pre-defined relations and the 31-40 relation types as unlabeled set with novel relations. We randomly selected 15% of the instances from the unlabeled set as the test set. The rest of the labeled and unlabeled instances are considered as the train set.

4.2 Compared Methods

To evaluate the effectiveness of our method, we select the following SOTA OpenRE models for comparison. Note that the first four methods are unsupervised and RSN as well as RSN-BERT leverages labeled data of predefined relations.

HAC with Re-weighted Word Embeddings (RW-HAC) (Elsahar et al., 2017). RW-HAC is a feature clustering method for OpenRE. The model contructs relational feature based on the weighted word embeddings as well as entity types.

Discrete-state Variational Autoencoder (VAE) (Marcheggiani and Titov, 2016). VAE is a reconstruction-based method for OpenRE. The model is optimized by reconstructing entities from pairing entities and predicted relations.

Entity Based URE (Etype+) (Tran et al., 2020). Etype+ is a simple and effective method relying only on entity types. The same link predictor as in

Hyper-parameters	value
optimizer	*Adam*
learning rate	1e-4
batch size	100
pre-training epochs L	10
BCE loss coefficient σ	2
center loss coefficient λ for FewRel	0.005
center loss coefficient λ for TACRED	0.001
ramp-up coefficient μ_0	1.0
ramp-up length T	10

Table 1: Hyper-parameter settings.

(Marcheggiani and Titov, 2016) is employed and two additional regularisers are used.

Self-supervised Feature Learning for OpenRE (SelfORE) (Hu et al., 2020). SelfORE exploits weak, self-supervised signals by leveraging large pretrained language model for adaptive clustering on contextualized relational features.

Relational Siamese Network (RSN) (Wu et al., 2019). This method learns similarity metrics of relations from labeled data of pre-defined relations, and then transfer the relational knowledge to identify novel relations in unlabeled data.

RSN with BERT Embedding (RSN-BERT). A variant of RSN, the static word vector is replaced by the BERT embedding for fair comparison.

4.3 Implementation Details

Our entity pair encoder is implemented as the `bert-base-uncased` which consists of 12 layers and we use layer 8 as the output layer for best performance. Note that we only fine-tune the parameters of the output layer in the iterative training process to avoid overfitting. Non-linear mapping $g(\cdot)$ and $d(\cdot)$ are both implemented as a DNN with `relu` activation, specifically $\mathbb{R}^d$-512-512-256 for $g(\cdot)$ and 256-512-512-$\mathbb{R}^d$ for $d(\cdot)$. All experiments are conducted using a GeForce GTX 1080Ti with 11GB memory and table 1 shows our best hyper-parameter settings.

5 Results and Analysis

In this section, we present the experimental results of our model on two real-world datasets to demonstrate the effectiveness of our method. We also provide additional experimental results on hyper-parameter analysis and relation representation visualization in appendix A and B.

Dataset	Method	B^3			V-measure			ARI
		Prec.	Rec.	F_1	Hom.	Comp.	F_1	
FewRel	VAE(Marcheggiani and Titov, 2016)	0.309	0.446	0.365	0.448	0.500	0.473	0.291
	RW-HAC(Elsahar et al., 2017)	0.256	0.492	0.337	0.391	0.485	0.433	0.250
	EType+(Tran et al., 2020)	0.238	0.485	0.319	0.364	0.463	0.408	0.249
	SelfORE(Hu et al., 2020)	0.672	0.685	0.678	0.779	0.788	0.783	0.647
	RSN(Wu et al., 2019)	0.486	0.742	0.589	0.644	0.787	0.708	0.453
	RSN-BERT	0.585	0.899	0.709	0.696	0.889	0.781	0.532
	RoCORE	0.752_{17}	0.846_{09}	$\mathbf{0.796}_{11}$	0.838_{10}	0.883_{06}	$\mathbf{0.860}_{07}$	$\mathbf{0.709}_{23}$
TACRED	VAE(Marcheggiani and Titov, 2016)	0.247	0.564	0.343	0.208	0.362	0.264	0.159
	RW-HAC(Elsahar et al., 2017)	0.426	0.633	0.509	0.469	0.597	0.526	0.281
	EType+(Tran et al., 2020)	0.302	0.803	0.439	0.260	0.607	0.364	0.143
	SelfORE(Hu et al., 2020)	0.576	0.510	0.541	0.630	0.608	0.619	0.447
	RSN(Wu et al., 2019)	0.628	0.634	0.631	0.624	0.663	0.643	0.459
	RSN-BERT	0.795	0.878	0.834	0.849	0.870	0.859	0.756
	RoCORE	0.871_{42}	0.849_{37}	$\mathbf{0.860}_{35}$	0.895_{32}	0.881_{20}	$\mathbf{0.888}_{24}$	$\mathbf{0.812}_{64}$

Table 2: Main results on two relation extraction datasets. The subscript represents the corresponding standard deviation (e.g., 0.796_{11} indicates 0.796 ± 0.011). Experimental results show that our method reduces the error rate by 29.2%($0.709 \rightarrow 0.796$) and 15.7%($0.834 \rightarrow 0.860$), on two datasets respectively.

5.1 Main Results

Table 2 reports model performances on FewRel, TACRED dataset, which shows that the proposed method achieves state-of-the-art results on OpenRE task. Benefitting from the valuable information in the labeled instances of pre-defined relations, RoCORE effectively learns the relation-oriented representation from which the derived clusters explicitly align with relational semantic classes, thereby outperforming previous clustering-based baseline such as SelfORE by a large margin. In addition, despite the fact that RSN and its variant RSN-BERT also leverage relational information in labeled data, the learning of similarity metrics and clustering are mutually independent. In our method, relation representation learning and cluster optimization are mutually dependent. Thus, the learned representations are tailored for clustering. As a result, our method outperforms RSN and RSN-BERT on the two datasets.

5.2 Ablation Study

To study the contribution of each component in the proposed method, we conduct ablation experiments on the two datasets and display the results in Table 3. The results show that the model performance is degraded if $\mathcal{L}_{center}$ is removed, indicating that the guidance of supervision signals from pre-defined relations provide valuable information for learning the relation-oriented representations. It is worth noting that the reconstruction term has an important role in the clustering module. Without

Dataset	Method	Prec.	Rec.	F_1
FewRel	w/o center loss	0.726_{32}	0.774_{31}	0.749_{31}
	w/o reconstruction	0.512_{38}	0.573_{17}	0.540_{25}
	w/o CE	0.662_{67}	0.787_{54}	0.719_{47}
	RoCORE	0.752_{17}	0.846_{09}	$\mathbf{0.796}_{11}$
TACRED	w/o center loss	0.818_{57}	0.842_{37}	0.830_{41}
	w/o reconstruction	0.549_{35}	0.483_{30}	0.514_{31}
	w/o CE	0.706_{45}	0.776_{51}	0.739_{47}
	RoCORE	0.871_{42}	0.849_{37}	$\mathbf{0.860}_{35}$

Table 3: Abalation study of our method. This table only lists the results of metric B^3. For results of other metrics, please refer to the Table 5 in Appendix C.

the reconstruction term, the semantic space will collapse and the performance will be seriously hurt. In addition, joint optimizing on both the labeled and unlabeled data is also very important. The initial pseudo labels for novel relations are not accurate due to the unwanted clustering bias on pre-defined relations. Without $\mathcal{L}_{CE}$, the error in pseudo labels will lead the refinement of the entity pair representation to a wrong direction, which affects the model performance.

5.3 The Influence of Pre-defined Relation Number on Performance

In this subsection, we conduct experiments on two different datasets to explore the influence of pre-defined relation number on performance of our method. For FewRel dataset, following the setting in (Wu et al., 2019), we change the number of pre-defined relations from 40 to 64 while fixing the total number of labeled instances to 25,000.
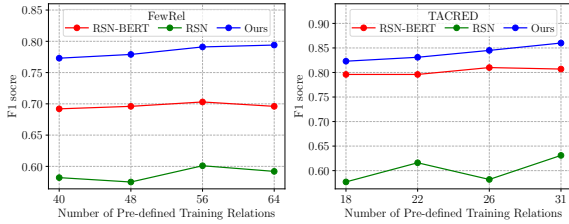
Figure 3: Clustering results with different numbers of pre-defined training relations.

Task	Method	Prec.	Rec.	F_1
	RSN	0.349	0.590	0.439
$\mathbf{F} \to \mathbf{T}$	RSN-BERT	0.337	0.866	0.486
	RoCORE	0.621_{28}	0.602_{51}	$\mathbf{0.611_{34}}$
	RSN	0.225	0.529	0.316
$\mathbf{T} \to \mathbf{F}$	RSN-BERT	0.261	0.861	0.400
	RoCORE	0.687_{36}	0.766_{46}	$\mathbf{0.724_{26}}$

Table 4: Results on two constructed cross-domain tasks. F means FewRel, which is from encyclopedia domain. T means TACRED, which is from news and web domain. This table only lists the results of metric B^3. For results of other metrics, please refer to the Table 6 in Appendix C.

Similarly, the settings for TACRED dataset is 18, 31 and 12, 000, respectively.

From figure 3 we can see the following: (1) The increase of pre-defined relation number do improve the generalization of our method on novel relations. The models trained on 64/31 relations slightly perform better than the models trained on 40/18 relations on FewRel/TACRED dataset (2) Our method constantly performs better than RSN and RSN-BERT with the number of predefined relations vary. This indicates the effectiveness of our method.

5.4 Cross Domain Analysis

In real-world settings, pre-defined relations and novel relations of interest usually come from different domains. To study the model performance in cross-domain settings, we conducted experiments on two cross-domain tasks, i.e,: FewRel to TACRED and TACRED to FewRel. Pre-defined relations and their labeled instances come from the source domain training dataset, and we evaluate performance on the target domain testing dataset.

Table 4 shows the experimental results, from which we can observe that: (1) the change of domain increases the semantic gap between the predefined and novel relations. As the result of that,
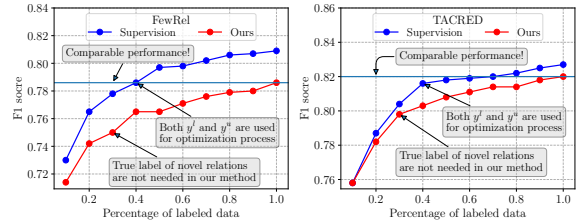

Figure 4: Model performance with different amounts of labeled data.

the performance of the model using labeled data of predefined relations is degraded. (2) compared with RSN and RSN-BERT, our method shows better generalization performance on novel relations, which shows that our proposed iterative joint training method effectively reduces the unwanted bias on source domain labeled data. (3) In addition, when a model has the tendency to cluster multiple relation into one, an unbalanced PR value (i.e., high *rec.* and low *prec.* in RSN-BERT) will be produced, which is undesired in real-world applications.

5.5 Incremental Learning of Novel Relations

In this subsection, we evaluate the effectiveness of our incremental learning scheme and explore the influence of the amount of labeled data on model performance. We use BERT with a linear softmax classifer as the baseline for comparison. We train the baseline model using the labeled data of both pre-defined and novel relations, following the supervised learning paradigm. For our method, we still use only the labels of pre-defined relations.

From figure 4 we can observe the following: (1) The performance of the models improve gradually as labeled data increase. Our method can still maintain good performance when there is a lack of labeled data. This indicates that the proposed method is robust to the reduction of labeled data. (2) Our method achieves similar performance compared with the supervised baseline on two experiments, which use 40% labels of novel relations on FewRel dataset and 82% on TACRED respectively. It indicates that we successfully achieve the incremental learning of novel relations.

6 Conclusions

In this work, we introduce a relation-oriented clustering method that extends the current unsupervised clustering-based OpenRE method. The proposed method leverages the labeled data of pre-defined relations to learn a relation-oriented representation

from which the derived clusters explicitly align with relational classes. Iterative joint training method effectively reduces the unwanted bias on labeled data. In addition, the proposed method can be easily extended to incremental learning of novel relations. Experimental results show that our method outperforms SOTA methods for OpenRE.

Acknowledgements

References

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 2787–2795, Red Hook, NY, USA. Curran Associates Inc.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of bert's attention. *CoRR*, abs/1906.04341.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Hady Elsahar, Elena Demidova, Simon Gottschalk, Christophe Gravier, and Frederique Laforest. 2017. Unsupervised open relation extraction. In *The Semantic Web: ESWC 2017 Satellite Events*, pages 12–16, Cham. Springer International Publishing.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. 2008. Open information extraction from the web. *Commun. ACM*, 51(12):68–74.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. 2020. Scan: Learning to classify images without labels.

Yoav Goldberg. 2019. Assessing bert's syntactic abilities. *CoRR*, abs/1901.05287.

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.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Xuming Hu, Chenwei Zhang, Yusong Xu, Lijie Wen, and Philip S. Yu. 2020. Selfore: Self-supervised relational feature learning for open relation extraction.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Diego Marcheggiani and Ivan Titov. 2016. Discrete-state variational autoencoders for joint discovery and factorization of relations. *Transactions of the Association for Computational Linguistics*, 4:231–244.

Matthew Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Brussels, Belgium. Association for Computational Linguistics.

Yusuke Shinyama and Satoshi Sekine. 2006. Preemptive information extraction using unrestricted relation discovery. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 304–311, New York City, USA. Association for Computational Linguistics.

Étienne Simon, Vincent Guigue, and Benjamin Piwowarski. 2019. Unsupervised information extraction: Regularizing discriminative approaches with relation distribution losses. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1378–1387, Florence, Italy. Association for Computational Linguistics.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. *CoRR*, abs/1906.03158.

Thy Thy Tran, Phong Le, and Sophia Ananiadou. 2020. Revisiting unsupervised relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7498–7505, Online. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Yijun Wang, Changzhi Sun, Yuanbin Wu, Hao Zhou, Lei Li, and Junchi Yan. 2021. ENPAR:enhancing entity and entity pair representations for joint entity relation extraction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2877–2887, Online. Association for Computational Linguistics.

Ruidong Wu, Yuan Yao, Xu Han, Ruobing Xie, Zhiyuan Liu, Fen Lin, Leyu Lin, and Maosong Sun. 2019. Open relation extraction: Relational knowledge transfer from supervised data to unsupervised data. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 219–228, Hong Kong, China. Association for Computational Linguistics.

Eric P. Xing, Andrew Y. Ng, Michael I. Jordan, and Stuart Russell. 2002. Distance metric learning, with application to clustering with side-information. In *Proceedings of the 15th International Conference on Neural Information Processing Systems*, NIPS'02, page 521–528, Cambridge, MA, USA. MIT Press.

Chenyan Xiong, Russell Power, and Jamie Callan. 2017. Explicit semantic ranking for academic search via knowledge graph embedding. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, page 1271–1279, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Limin Yao, Aria Haghighi, Sebastian Riedel, and Andrew McCallum. 2011. Structured relation discovery using generative models. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1456–1466, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Alexander Yates, Michele Banko, Matthew Broadhead, Michael Cafarella, Oren Etzioni, and Stephen Soderland. 2007. TextRunner: Open information extraction on the web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 25–26, Rochester, New York, USA. Association for Computational Linguistics.

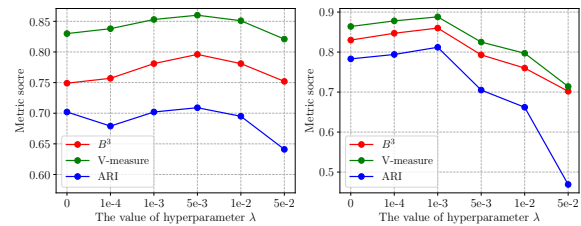Mo Yu, Wenpeng Yin, Kazi Saidul Hasan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2017.

Figure 5: Model performance with different λ.

Improved neural relation detection for knowledge base question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 571–581, Vancouver, Canada. Association for Computational Linguistics.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark. Association for Computational Linguistics.

A Hyperparameter Analysis

From the experimental results of ablation study, it can be seen that reconstruction loss and center loss have a great impact on the performance of the model. λ is a key hyperparameter that balances the reconstruction loss versus center loss. In this section, we conduct experiments to study the influence of the value of λ on the performance of the model. From Figure 5 we can see that: (1) When λ gradually increases from 0, the center loss begins to affect the optimization. The model learns that instances with the same relation should be mapped to relatively close positions in the representation space, and the performance of the model gradually improves. (2) When the lambda exceeds a certain threshold, further increasing the λ will leads to unwanted bias to the predefined relations, which will degrade the performance of the model.

B Relation Representation Visualization

To intuitively show how the RoCORE method learns the constantly optimized relation-oriented representation, we visualize the relational representation with t-SNE (van der Maaten and Hinton, 2008). The visualization results are shown in Figure 6. It is apparent that, before training (left), the relational representations are distributed randomly at different locations in the semantic
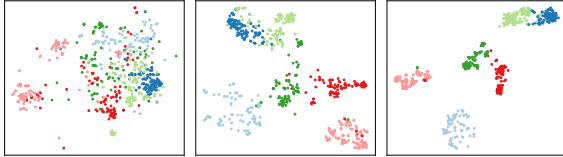
Figure 6: Visualization of the relation representation after t-SNE dimension reduction. The representations are colored with their ground-truth relation labels. These three sequentially illustrate the feature representation of initial sate, after reconstruction pre-training, and after training. All figures visualize the clustering result for 600 instances of randomly selected 6 novel relations on FewRel test dataset.

space. After pre-training (middle), the relational representations still are not tailored for the relations. For example, the instances with blue and light green colors may have similar syntactic or surface features and clustering them directly will lead to a poor result. After training (right), the relational representations are well separated and the distribution is based on relation types.

C Detailed Results of Other Experiments

In this section, the detailed results of ablation experiments and cross domain analysis are listed in Table 5 and Table 6 respectively.

Dataset	Method	B^3			V-measure			ARI
		Prec.	Rec.	F_1	Hom.	Comp.	F_1	
FewRel	w/o center loss	0.726_{32}	0.774_{31}	0.749_{31}	0.818_{19}	0.842_{18}	0.830_{19}	0.702_{38}
	w/o reconstruction	0.512_{38}	0.573_{17}	0.540_{25}	0.665_{24}	0.689_{12}	0.676_{16}	0.495_{40}
	w/o CE	0.662_{67}	0.787_{54}	0.719_{47}	0.772_{41}	0.844_{27}	0.806_{28}	0.617_{68}
	RoCORE	0.752_{17}	0.846_{09}	$\mathbf{0.796_{11}}$	0.838_{10}	0.883_{06}	$\mathbf{0.860_{07}}$	$\mathbf{0.709_{23}}$
TACRED	w/o center loss	0.818_{57}	0.842_{37}	0.830_{41}	0.855_{41}	0.867_{32}	0.864_{32}	0.783_{69}
	w/o reconstruction	0.549_{35}	0.483_{30}	0.514_{31}	0.589_{37}	0.570_{28}	0.579_{32}	0.393_{54}
	w/o CE	0.706_{45}	0.776_{51}	0.739_{47}	0.753_{30}	0.803_{37}	0.777_{32}	0.656_{85}
	RoCORE	0.871_{42}	0.849_{37}	$\mathbf{0.860_{35}}$	0.895_{32}	0.881_{20}	$\mathbf{0.888_{24}}$	$\mathbf{0.812_{64}}$

Table 5: The detailed results of abalation study. The subscript represents the corresponding standard deviation (e.g., 0.749_{12} indicates 0.749 ± 0.012)

Task	Method	B^3			V-measure			ARI
		Prec.	Rec.	F_1	Hom.	Comp.	F_1	
F → T	RSN	0.349	0.590	0.439	0.387	0.533	0.448	0.279
	RSN-BERT	0.337	0.866	0.486	0.400	0.777	0.528	0.352
	RoCORE	0.621_{28}	0.602_{51}	0.611_{34}	0.642_{37}	0.666_{32}	0.654_{31}	0.451_{65}
F → T	RSN	0.225	0.529	0.316	0.359	0.507	0.420	0.243
	RSN-BERT	0.261	0.861	0.400	0.438	0.822	0.571	0.263
	RoCORE	0.687_{36}	0.766_{46}	$\mathbf{0.724_{26}}$	0.796_{22}	0.836_{22}	$\mathbf{0.815_{16}}$	$\mathbf{0.658_{43}}$

Table 6: The detailed results of cross domain analysis. The subscript represents the corresponding standard deviation (e.g., 0.724_{26} indicates 0.724 ± 0.026)