Cognitive Computation
https://doi.org/10.1007/s12559-024-10338-5

Scrutinizing Label: Contrastive Learning on Label Semantics and Enriched Representation for Relation Extraction

Zhenyu Zhou[1,2,3,4] · **Qinghua Zhang**[2,3,4] · **Fan Zhao**[1,2,3,4]

Received: 19 April 2023 / Accepted: 30 July 2024

Abstract
Sentence-level relation extraction is a technique for extracting factual information about relationships between entities from a sentence. However, the customary method overlooks the semantic information conveyed by the label itself, thereby compromising the efficacy of rare types. Furthermore, there is a growing interest in exploring the use of textual information as a crucial resource to enhance RE models for more effectiveness. To address these two issues, CLERE (*C*ontrastive *L*earning and *E*nriched Representation for *R*elation *E*xtraction) based on contrastive learning and enriched representation of context is proposed. Firstly, by contrastive learning to incorporate semantic information of labels, CLERE is able to effectively convey and exploit the underlying semantics of various sample categories. Thereby enhancing its semantics understanding and classification capabilities, the issue of misclassification due to data imbalance is alleviated. Secondly, both semantics of context and positional information of tagged entities are enhanced by employing weighted layer pooling on pre-trained language models, which improves the representation of context and entity mentions. Experiments are conducted on three public dataset to authenticate the effectiveness of CLERE. The results demonstrate that the proposed model outperforms existing mainstream baseline methods significantly.

Keywords Sentence-level relation extraction · Contrastive learning · Semantic similarity · Pre-trained language models

Introduction

Relation extraction (RE) is a crucial component of natural language processing (NLP) and serves as a vital link between downstream tasks, such as event extraction (EE) [1] and knowledge graph construction, and upstream tasks, such as named entity recognition (NER) [2, 3] and entity linking (EL) [4]. Based on a predefined set of relationships, the objective of RE is to identify the relationships between two entities within a given text. Three instances are depicted in Fig. 1, where the subjects and objects are marked in the sentence part, and the labels of the three instances belong to three different categories.

With the development of pre-trained language models (PLMs) based on the Transformers architecture [5–7]. Supported by a large training corpus, PLMs have shown remarkable performance in representing long sentences across a diverse array of NLP tasks. Particularly in supervised sentence-level RE, researchers have proposed models that incorporate PLMs, their performance far superior to those based on recurrent neural networks (RNNs) and convolutional neural networks (CNNs) [8–11]. Leveraging the information available in the dataset is the key step of the RE task. The focus of most primary works has been to develop efficient ways to utilize the textual information of a sentence [12]. To this end, entity masking [10] has been proposed as a technique to leverage the entity information present in the text, and its effectiveness has been remarkable. However, researchers have overlooked the wealth of semantic information conveyed by labels, which can differ significantly across

✉ Qinghua Zhang
zhangqh@cqupt.edu.cn

1 School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

2 Key Laboratory of Big Data Intelligent Computing, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

3 Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

4 Key Laboratory of Tourism Multisource Data Perception and Decision, Ministry of Culture and Tourism, Chongqing 400065, China

Published online: 22 August 2024

Sentence	Relation
Billy Mays the bearded, ..., died at his home in **Tampa**, ...	per:city of death
... **Bibi** confessed to committing ... with the **Muslim** women.	no_realtion
... giant **Countrywide Financial** on Friday, coupled with expectations for a **US** ...	org:country of headquarters

Fig. 1 An example including three types of relations of RE task

different categories of labels [13]. The semantic information conveyed by labels plays a crucial role if there is a data imbalance. To investigate whether different labels carry significant differences in their inherent textual semantics, which can be used to strictly differentiate them from each other, the textual semantics of the dataset's label set is calculated by SBERT [14]. The semantic similarity among the 11 relations in the TACRED dataset is demonstrated in Fig. 2. Based on Fig. 2a, it can be observed that both "per" and "org" type labels have a low semantic similarity to the "no relation" category. Furthermore, Fig. 2b illustrates the semantic similarity between the "per" and "org" types, which is also relatively small. It can be inferred from this experiment that there exist significant dissimilarities in the textual semantics of the labels.

Noteworthy, both Nayak et al. [15] and Mondal et al. [16] noticed that the context category is essential to embody the sentences. Peng et al. [17] have noted that the model may acquire certain surface information of the dataset through entity mentions, thereby impeding the model's contextual comprehension. The researchers evaluated the efficacy of incorporating contextual information with entity mentions in their approach and contrasted it with approaches that solely relied on either entity mentions or contextual information. Empirical evidence [18] from the classification task demonstrated that the amalgamation of context and entity mentions outperformed the other two methodologies. Therefore, in this paper, a fusion of context and entity mention is adopted, while simultaneously utilizing weighted layer pooling to augment the contextual representation, thereby maximizing the use of the information conveyed by the sequences themselves.

The works of SimCSE [19] show contrastive learning has made a splash in unsupervised text classification, a method that can accurately capture sample differences. Supervised contrastive learning [20] proposes a loss that introduces contrastive learning from the unsupervised domain into the supervised field; the connection between supervised contrastive loss and the triplet loss is also explored.

Peng et al. [17] redesign the pre-training task of PLMs with the aim of ensuring that sentences sharing similar relations exhibit analogous representations, while those with different relations manifest distinct disparities. This innovative approach combines contrastive pre-training task with masked language modeling task in the overall model training objective. The clear advantage of this methodology is its partial bridging of the gap between PLM pre-training tasks and relation extraction tasks. However, it comes at the cost of increased computational resources and suffers from low reusability, necessitating the re-pre-training of the entire model for different PLM types, rendering it unsuitable for modular application across multiple domains.

CLERE (*C*ontrastive *L*earning and *E*nriched representation for *R*elation *E*xtraction) opts against full re-pre-training, instead integrating comparative learning directly into the training phase. Moreover, to address the gap between

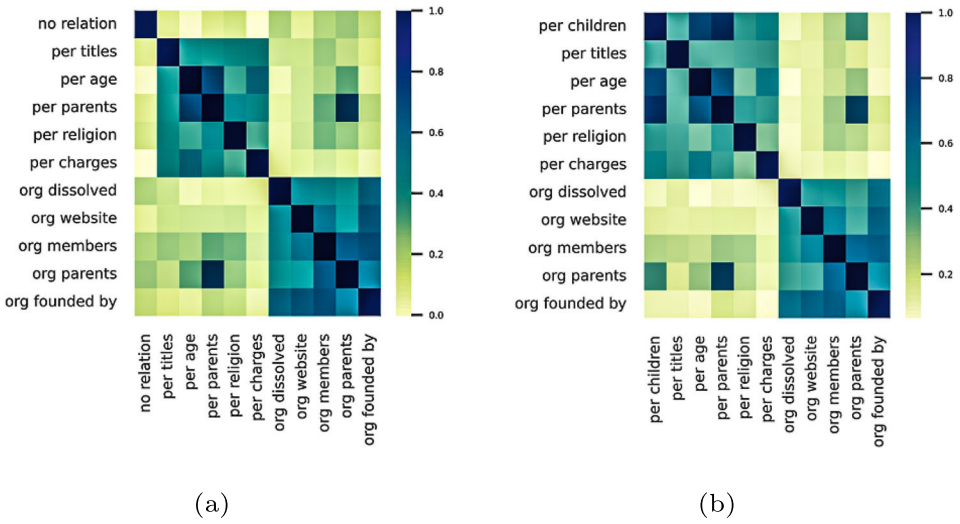

Fig. 2 Text similarity between the relationships from TACRED dataset. **a** Including "no relation" category. **b** Not including "no relation" category

pre-training and RE tasks, CLERE considers refined strategies for selecting input sentences and labeling entity embeddings. Additionally, it delves into the structural intricacies of PLMs at each layer, analyzing their compositions and scrutinizing the combination of embeddings obtained at various layers to yield a more nuanced and semantically rich representation. During inference, the closest one to the sequence from the candidate labels is selected as the final result. CLERE enhances the performance of existing RE models and incorporates the valuable information provided by labels, making it particularly effective in scenarios where there is data imbalance.

To assess the efficacy of CLERE, experiments are conducted on three supervised RE dataset, utilizing BERT-base and Roberta-large as PLMs, respectively. The outcomes demonstrated that the baseline models are surpassed by CLERE. Additionally, the reasons behind the model's success are investigated. To summarize, our contributions can be outlined as follows.

- The combination of semantic information of labels with contextual information is explored, and considerable differences in the semantic information between the different labels are found. Additionally, the role of pooling strategies in generating contextual semantic embeddings is explored.
- A relation extraction model that enhances PLMs embedding ability and applies the concept of contrastive learning to leverage label semantic embeddings is proposed. This brings the semantics of the text closer to the positive labels and moves away from the negative ones.
- Experiments are done on three public dataset on which CLERE achieved above-baseline results, and higher recall and higher F1 scores are achieved when using the same PLMs.

Related Work

Supervised RE

Supervised RE is a well-researched area within NLP, and early methods used primarily feature-based and kernel-based approaches. Feature-based approaches [21] involve the design of features for entities and their corresponding contexts, including lexicon, syntax, and semantics, which are then fed into an entity-relation classifier. With the advent of SVM, kernel-based approaches have also received considerable attention, with kernel functions designed to obtain similarities between relation representations and text instances. However, feature-based methods heavily rely on manually crafted features, which require researchers to possess domain-specific background knowledge. Kernel-based methods require the use of natural language processing toolkits to transform input text into syntactic dependency trees, which can result in a relatively high probability of error propagation.

Deep learning-based methods have also been used for supervised RE. Liu et al. [22] were one of the first to use CNNs for this task, but this method still requires the use of NLP toolkits. The idea of entity position embedding was introduced by Zeng et al. [23], which later served as the foundation for entity awareness. However, the use of fixed-size convolution kernels in this method resulted in the loss of global features. To address this issue, Nguyen et al. [24] utilized convolution kernels of multiple sizes, which focused on both local and global features. Zhang et al. [25] employed Bi-RNN for RE tasks. To mitigate the problem of RNN gradient explosion, Xu et al. [26] proposed a model with an LSTM structure, which proved effective in extracting sentence-level features in RE tasks. However, the features extracted by this model are still insufficient to achieve optimal performance.

With the advent of PLMs, the landscape of NLP has been revolutionized. This progress has been propelled by the introduction of the Transformer architecture, which features a self-attention mechanism. Among these models, BERT [6], trained on a large corpus, has demonstrated an unparalleled ability to capture textual features. The majority of RE that are based on PLMs utilize BERT or one of its variants as a PLM [27]. These models can be broadly classified into two main categories. One is to revamp the pre-training task by enhancing the internal structure of BERT. Roberta [7] uses a larger dataset and a novel dynamic masking technique to provide a higher level of understanding of sentence context, leading to better performance on multiple NLP tasks including RE. KnowBERT [28] introduces an external knowledge base and improves the training objectives of BERT by constructing the entities in the knowledge base as a triad, thus achieving even more advanced performance in text understanding. LUKE [29] has made a significant breakthrough in the field of PLMs by enhancing entity perception on top of BERT. By incorporating entity types and attributes into the representation process, LUKE has achieved performance that outperforms BERT on tasks such as entity perception and question-answering systems. In general, the advantage of this approach lies in its ability to facilitate the learning of task-specific language representations in a more directed manner, thus circumventing overfitting during subsequent fine-tuning and enhancing the generalization performance of the model. Nonetheless, the downside of this method is apparent: the redesign of the pre-training task is computationally expensive and poses greater demands on the model structure and training process design. Moreover, pre-training tasks that are tailored to specific domains may only be suitable

for a particular task and cannot be extrapolated to other tasks. The second approach, fine-tuning, is widely employed today. In this approach, task-specific components are added to the PLMs, enabling the model to achieve advanced performance without requiring further pre-training. R-BERT [11] is an advanced model for relational extraction, based upon the mighty BERT architecture. It enhances BERT's ability to model relationships by introducing token-level relational representations. MTB [10] suggests that using partial embeddings of entities can achieve even better entity representation for RE. REDN [30] argues that the relationship is determined by the relevance of the subject and object entities, and the representation of the relationship should be a matrix rather than a one-dimensional array. A corresponding loss function is also proposed in REDN. The advantage of this approach is that high performance can be achieved by only designing fine-tuning modules for PLMs while consuming fewer computational resources. The disadvantage is that a large training dataset specific to the task is required, which must have significantly more domain-specific properties than the pre-trained dataset. Additionally, the fine-tuning model is prone to overfitting when the fine-tuning dataset differs significantly from the pre-training dataset.

Contrastive Learning

Contrastive learning has become a mainstream unsupervised learning method in recent years. It is assumed that there is a semantic relationship between α_i and α_i^+; let R_i and R_i^+ serve as representations of α_i and α_i^+. With a mini-batch of N-pairs (α_i, α_i^+), the training goal is

$$ConLoss = -\log \frac{e^{\text{sim}\left(\mathbf{R}_i, \mathbf{R}_i^+\right)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}\left(\mathbf{R}_i, \mathbf{R}_j^+\right)/\tau}}, \quad (1)$$

where τ is a temperature hyperparameter as well as sim (h_1, h_2) represents the cosine similarity $\frac{\mathbf{R}_1^\top \mathbf{R}_2}{\|\mathbf{R}_1\| \cdot \|\mathbf{R}_2\|}$.

Hadsell et al. [31] proposed an algorithm for "learning comparable distances," which maps samples of the same category to a tight space and samples of different categories to a more distant space. Contrastive learning methods have evolved and introduced contrastive loss, which learns discriminative feature representations by minimizing the distance of similar samples and maximizing the distance of dissimilar samples. In the field of NLP, contrastive learning methods such as RankCSE [32], SimCSE [19], and BERT-CL [33] have been used to address text similarity matching problems and enhance the representation of BERT through contrastive learning, respectively. In the RE field, Peng et al. [17] propose a contrastive learning framework with entity mention, where examples that are defined to be adjacent are clustered together and those that are not are pushed apart. This model's training objective combines the contrastive learning objective with the masked language modeling objective. Finally, Chen et al. [34] apply the contrastive learning idea to remotely supervised relational extraction, further demonstrating the versatility and potential of contrastive learning in NLP.

Furthermore, Khosla et al. [20] have noted that triplet loss represents a particular instance of contrastive loss, specifically when only one positive and one negative sample are utilized. Unlike the standard contrastive loss, triplet loss operates on triplets, consisting of an anchor, a positive, and a negative sample. Its objective is to minimize the distance between the anchor and the positive sample, while simultaneously maximizing the distance between the anchor and the negative sample. Consequently, triplet loss aims to ensure that the distance between the anchor and positive samples is smaller than that between the anchor and negative samples by at least a specified margin, failing which incurs a loss penalty. In contrast, N-Pair Loss emerges as a more suitable solution for addressing scenarios involving 1 positive and N negative, N positive and N negative. As an extension of triplet loss, N-Pair Loss harnesses the advantages of leveraging information from multiple negative samples in each update iteration, aiming to guarantee that the embedding of the current sample is distinctly distant from all types of negative samples. However, when the quantity of negative samples is substantial, the model may encounter challenges in convergence or might become susceptible to local optima. Moreover, the computational complexity of N-Pair Loss escalates exponentially compared to triplet loss since it necessitates computing the similarity score between the anchor sample and all negative samples.

CLERE

Overview

The model framework is illustrated in Fig. 3. The training and inference process of CLERE can be divided into two steps; in step 1: Training process, the input of the model will be divided into three parts: sentences, positive labels, and negative labels. The position of entities in sentences will be marked using entity mention, which will be added by placing special markers("#" and "$") around the entities. The positive instance label refers to the original label of the instance, while the negative instance label is randomly selected from a set of labels, both of which are text-based. PLMs are employed to encode these three parts. The instance that is encoded will serve as the anchor, the positive label will be treated as the "pos" term, and the negative label will be treated as the "neg"

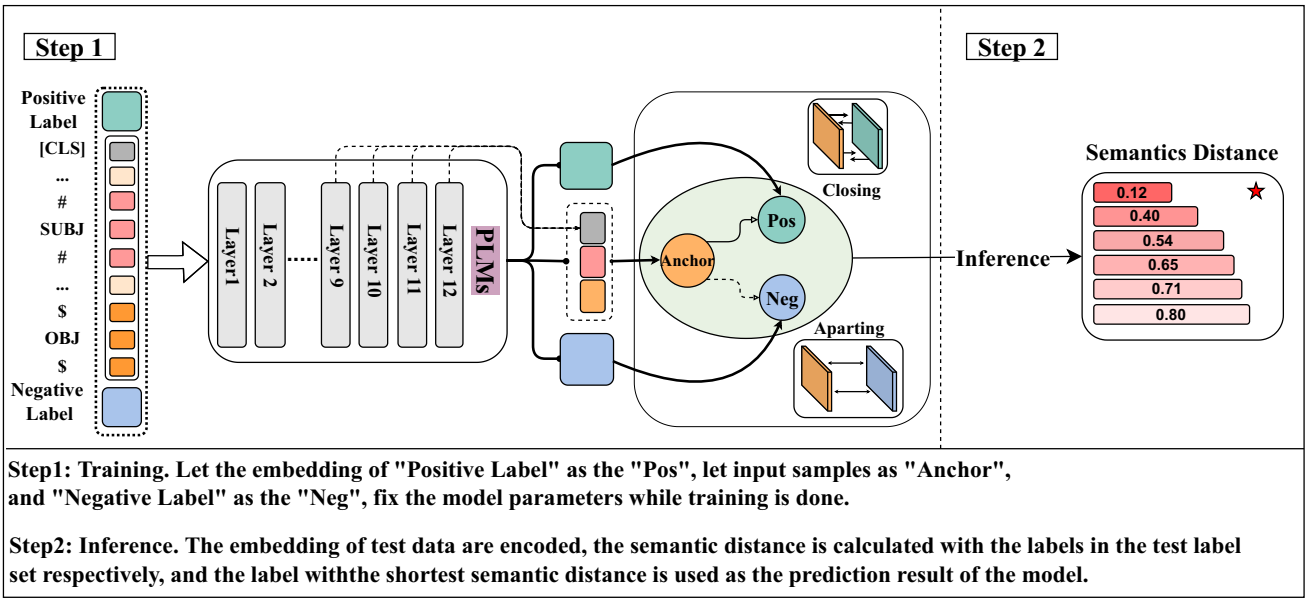

Fig. 3 The framework of CLERE consists of an input layer, an embedding layer, a loss learning layer, and an inference layer

term. The objective of the training is to minimize the distance between the anchor and the "pos" term while maximizing the distance between the anchor and the "neg" term. During the step 2: Inference process, the model selects the label with the shortest distance to the anchor as the prediction.

Problem Descriptions

Supervised RE at the sentence level is focused. Specifically, given an instance that contains the sentence *X*, the location and the entity type of *Subject* and *Object*, the task is to determine which predefined relation the entity pair belongs to. In other words, this is a classification task that aims to select the most appropriate label from a set of predefined relationship types.

Input Embedding

Sentence Embedding

In contrast to other NLP tasks, sentence embedding for RE is focused on maximizing the amount of information pertaining to the entities within the sentence. Building on previous work [10], the Typed Entity Markers (punct) [8] is utilized to represent the entities. Specifically, the "#" marker denotes the subject entity, and the "$" marker denotes the object entity. Additionally, the entity type information in textual form uses "$*$" and "$\wedge$" to mark the position of the entity type. To capture the sequence semantics of the sentence, the "[CLS]" and "[SEP]" are also added. The final input sentence to the PLMs takes the following form: [CLS]... # $*$ subj_type $*$ SUBJ #... $ $\wedge$ obj_type $\wedge$ OBJ $.... [SEP] (Fig. 4).

After pre-processing the input sentences, they are fed into PLMs to obtain embeddings. It is suggested by Peng et al. [17] that sufficient embedding information for RE can be provided by combining sentence contextual embedding with entity mention, while using only entity mention may lead to shortcuts in the model. Therefore, combining contextual embedding with the entity mention in this study. It is common practice to use the embedding of "[CLS]" when obtaining the sentence context. However, the contextual embedding obtained using this method is flawed, as it cannot capture the complete semantics of the sentence; the advantages and disadvantages of this approach will be discussed in Section "Experiments Analysis." In this paper, the preprocessed parts of the sentence are initially input into the PLM to generate a set of hidden states X for layers of the PLM as

$$X = PLM(sentence). \tag{2}$$

Input sentence

[CLS] At the same time, Chief Financial Official **#** * PERSON * **Douglas Flint** **#**, will become **$** ^ TITLE ^ **chairman** **$**, succeeding Stephen Green who is leaving to take a government job **[SEP]**

Fig. 4 An example of an input sentence using typed entity markers (punct)

Then, the weighted layer pooling [35] method is utilized to obtain contextual embeddings as

$$Seq = \frac{\sum_{i=1}^{N} \omega_i x_i}{\sum_{i=1}^{N} \omega_i}, \tag{3}$$

where $x_i \in X, i \in (1, N)$, and N refers to the number of layers selected from the PLM; in this paper, $N = 4$. The weight parameter ω_i is a learnable parameter that is initialized as a random matrix drawn from a uniform distribution.

After obtaining the contextual embeddings of the sequence through weighted layer pooling, next further extract embeddings H_1 and H_2 from the two entities mentioned in the sentence and then concatenated them with the contextual embedding sequence. The concatenated embeddings are input into a fully connected layer, followed by an activation function. Then, the anchor can be obtained An as

$$An = LeakyReLU(W_f(concat(Seq, H_1, H_2))), \tag{4}$$

where $W_f \in \mathbf{R}^{d \times 3d}$ (d is the hidden state size of PLMs).

Label Embedding

During the training step, to obtain the embeddings of the labelled text, the label is treated as normal text, i.e., remove the special characters("/" and "_") from the labels. Regarding the selection of positive and negative label pairs, the label of the sample itself is selected as the positive label $y \in Y$ and randomly select a label $\tilde{y} \in Y/y$ as the negative label. The label pair $[y, \tilde{y}]$ is then fed into PLMs to obtain the label embedding as

$$Pos, Neg = PLMs(y, \tilde{y}). \tag{5}$$

Training Objective

After the An, Pos, and Neg embedding matrices are obtained, the triplet loss function is utilized to minimize the semantic distance between the input and its positive label, while increasing the distance between the input and the negative label in the embedding space. It is achieved by applying the triplet loss function, thus enhancing the semantic correlation between the An and the Pos during the inference step, as illustrated in Fig. 5a. The distance between the An and the Pos is referred to as pos_dist, while the distance between the An and the Neg is referred to as neg_dist. Figure 5b depicts the variation of loss with respect to pos_dist and neg_dist, and it is evident that when pos_dist is at its minimum and neg_dist is at its maximum, the loss will decrease.

In the case of an anchor accompanied by its corresponding positive and negative instances, the function is mathematically formulated as exemplified in (7). The computation of the cosine similarity between the An and Pos and the cosine similarity between the An and Neg is performed as (6):

$$\mathrm{C(An, Pos)} = \frac{An \cdot Pos}{\|An\|\|Pos\|}, \mathrm{C(An, Neg)} = \frac{An \cdot Neg}{\|An\|\|Neg\|}. \tag{6}$$

The incorporation of a margin parameter to regulate the degree to which the distance between the An and Pos is smaller than that between the An and its Neg, thus preventing the model from taking shortcuts. During the training process, the objective of the model is to minimize the triplet loss and acquire the optimal embedding approach for the given data.

$$\mathcal{L}_{a,p,n} = \max(C(An, Pos) - C(An, Neg) + \mathrm{margin}, 0). \tag{7}$$

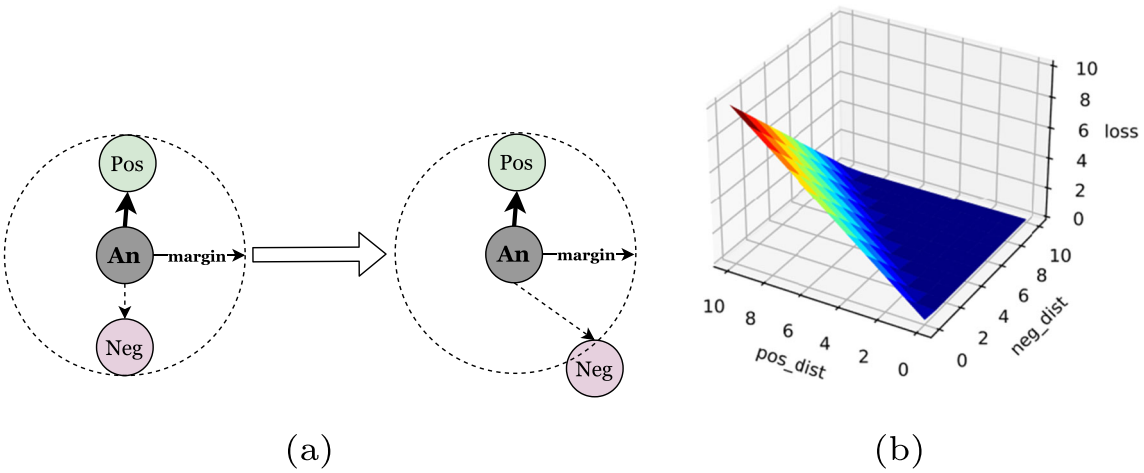

Fig. 5 **a** Triplet loss. It seeks to push Neg outside the circle defined by the margin and pull Pos inside. **b** Loss function display

The comprehensive training objective of the model is demonstrated in (8):

$$\mathcal{L}_{all} = \frac{1}{N_t} \sum_{a \in A} \sum_{p \in P_t} \mathcal{L}_{a,p,n}, \tag{8}$$

where A refers to the assemblage of training dataset, P_t denotes the set of positive labels, and N_t represents the total count of unique pairs comprising the training sentences and their corresponding affirmative labels.

In the inference phase, as the number of relationships in the dataset is fixed, no relations in the validation and test sets ever appeared in the training set. By utilizing the semantic similarity calculation of the model, the closest one to anchor from the set of labels in the test set is employed as the prediction result. To better illustrate the steps of the CLERE task, Algorithm 1 shows the peculiarities of training and inference for the CLERE.

Experiments

This section introduces the dataset being implemented, alongside the experimental parameter configurations, metrics, and baseline model against which comparisons are made. Subsequently, the experimental of the proposed approach is presented. Finally, the analysis of the observed outcomes is summarized.

Algorithm 1 The training and inference process of CLERE.

Input: Sentence $s \in S$, label $y \in Y$, $\tilde{y} \in Y/y$;
Output: A set of relationships $Result = \{r_1, r_2, r_3, ..., r_n\}$;
Initialize: Sequence embedding seq,
Initialize: Entity embedding $train.embed_1, train.embed_2$;
$Result \leftarrow \Phi$;
$An, Pos, Neg \leftarrow \Phi$;
$dist \leftarrow \Phi$;
for s **in** S
  seq = WLP(BERT(s)); // based on (3)
  $embed_1, embed_2$ = PLMs(s)); // based on (5)
  $An = concat(seq, embed_1, embed_2)$;// // based on (4)
  Pos, Neg = PLMs($y, \tilde{y}$); // based on (5)
  Tripletloss(An, Pos, Neg); // based on (7)
end for
for y **in** Y // Inference step
  $test = concat(seq_{test}, test.embed_1, test.embed_2)$;
  $inf\_embed, y\_embed = PLMs(test), PLMs(y)$;
  $dist$ = 1 - CosSim(inf_embed, y_embed);
  r = Min($dist$);
  $Result = Result + r$;
end for
return $Result$;

Table 1 Statistics of different dataset

Dataset	#train	#dev	#test	#rel	#no_relation[1]
TACRED	68,124	22,631	15,509	42	79.5%
TACREV	68,124	22,631	15,509	42	79.8%
RE-TACRED	58,465	19,584	13,418	40	63.2%

[1] Percentage of whole dataset occupied by no_relation

Dataset

Three versions of the TACRED [36] dataset will be used to evaluate CLERE: the original *TACRED* dataset, the *TACREV* [37] dataset, and the *RE-TACRED* [38] dataset. The particulars regarding those dataset can be observed within Table 1.

With 42 relations (including "no_relation"), the TACRED[1] dataset is one of the most extensive dataset to be used for supervised RE, and it is worth noting that the absolute majority of relationships in the dataset are "no_relation."

TACREV[2] is a modified version of the TACRED, where some of the errors in the validation and test sets of the TACRED dataset have been corrected, while the training set remains unchanged. Forty-two relations are retained in TACREV.

RE-TACRED[3] is another version of the TACRED dataset that complements some of the shortcomings of the original version by reconstructing the training, validation, and test blocks of the original version. RE-TACRED is even more so with only 40 relations.

Baselines

To assess the effectiveness of CLERE, a diverse set of existing approaches is compared with CLERE, including CNN-based [22, 23], RNN-based, GCN-based, and Transformers-based methods. *PA-LSTM* [36] combines a bi-directional LSTM sequence model with entity location-aware attention. *C-GCN* [39] utilizes GCNs [40] to encode sentences with dependency structures and predicts relations based on them. The C-GCN shows that dependency-based and sequence-based models have a complementary role. *SpanBERT* [41] is a PLM that builds on BERT [6] by enhancing the masking process for contiguous entities, removing the next sentence prediction (NSP) task, and introducing the span boundary objective (SBO) training target. These improvements have resulted in a significant enhancement over BERT for extractive tasks. *KnowBERT* [28] improves upon BERT by integrating multi-

[1] https://catalog.ldc.upenn.edu/LDC2018T03, Feb. 2023.

[2] https://github.com/DFKI-NLP/tacrev, Feb. 2023.

[3] arXiv:2104.08398, Feb. 2023.

Table 2 Hyperparameters in experiments

Parameter	Bert-base	Roberta-large
Batch_size	32	16
lr	3e-5	5e-6
Adam_epsilon	1e-6	1e-6
Drop_rate	0.1	0.1
Epoch	100	100
Margin	0.1	0.1
Max_seq_length	384	512
Max_label_length	20	20

ple real-world knowledge bases into its pre-training process, with the aim of enhancing its coverage of real-world knowledge. As a result, KnowBERT exhibits superior performance in downstream tasks such as entity extraction, RE, and disambiguation. *LUKE* [29] is a specialized representation designed for entity-related tasks that incorporate an entity-aware self-attention mechanism. This attention mechanism allows the model to focus more on the entities in the corpus, resulting in superior performance on downstream tasks. Both *MTB* [10] and *RIB* [8] demonstrate that the quality of representations generated by PLMs can be further improved by utilizing only the links between entities.

Model Configuration and Metrics

To ensure a fair comparison with other models, the official publicly available code provided in the papers is utilized, while adhering to the recommended hyperparameters. CLERE is implemented using the Transformers package of HuggingFace,[4] and training is performed using the Adam optimizer. All experimental results are reported as the average of 5 experiments using different random seeds. The details of the experimental hyperparameters are shown in Table 2.

To evaluate the performance of CLERE, Mirco F1 (11) is adopted as a metric, which is a common metric used in previous works. Mirco F1 takes into account the precision (9) and recall (10) of the classifier and is used to evaluate the overall performance of multi-classification problems. Mirco F1 computes the F1 score for each class and then averages them by weighting the number of samples in each class. This ensures that each class has an equal impact on the overall results.

$$Precision = \frac{\sum_i TP_i}{\sum_i (TP_i + FP_i)}, \tag{9}$$

$$Recall = \frac{\sum_i TP_i}{\sum_i (TP_i + FN_i)}, \tag{10}$$

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}. \tag{11}$$

Main Results

Table 3 shows the results of CLERE on TACRED and TACREV dataset. Table 4 shows the results of CLERE on RE-TACRED dataset.

CLERE outperforms these baselines on two of the dataset, achieving the same level of results as the current state-of-the-art on the last dataset, including attaining an F1-score of 74.9% on the TACRED dataset, 83.9% on the TACREV dataset, and 91.1%on the RE-TACRED, which is comparable with the current state-of-the-art. Superior performance is still achieved by CLERE without any extended dataset and further pre-training steps being used when contrasted with models such as SpanBERT, KnowBERT, and LUKE among the many Transformers-based methods. To validate the robustness of CLERE, even with a modest-sized PLM, BERT-base is selected as the PLM and performs experiments on the identical dataset comparing those baselines using the same PLM (including KnowBERT and SpanBERT); the comparison on the TACRED and TACREV dataset is still outperformed by CLERE and achieves the highest recall in RE-TACRED.

In addition, the results show that CLERE has a higher recall compared to the fine-tuned models when using the same PLM, which validates our conjecture of introducing label information into the training process and using the idea of contrastive learning to solve data imbalance. The models that use retraining and the fine-tuned models have achieved higher precision beyond CLERE, but lower recall and F1 score, again illustrating the advanced nature of CLERE. Overall, the efficacy of CLERE has been well demonstrated by the above experimental results.

Experiments Analysis

In this section, an analysis will be conducted to determine why the model's pooling strategy may have a positive impact on the model's performance. Furthermore, the performance of CLERE on an unbalanced dataset and the selection of the margin parameter in the triplet loss function will be discussed. Lastly, a case study will be presented to illustrate the inference steps of CLERE.

Ablation Experiments

The structure of transformers is frequently fine-tuned by incorporating an additional output layer for downstream tasks or models. The final layer of representations of PLMs

[4] https://huggingface.co, Feb. 2023.

Table 3 Precision, recall, and F1(in %) on TACRED and TACREV dataset

Models	TACRED			TACREV		
	Precision	Recall	F1	Precision	Recall	F1
CNN − based						
CNN¬	50.0	32.6	39.4	51.8	35.9	42.4
CNN+BERT$^{¬}_{base}$	71.9	51.1	59.7	79.5	60.2	68.5
RNN − based						
Bi-LSTM¬	53.3	57.5	55.7	58.6	67.7	62.6
Bi-LSTM+BERT$^{¬}_{base}$	65.3	59.9	62.5	71.8	70.2	71.0
PA-LSTM† [36]	68.1	64.5	70.1	–	–	73.3
GCN − based						
GCN¬	65.6	50.5	57.1	72.4	59.3	65.2
GCN+BERT$^{¬}_{base}$	66.3	58.8	62.4	73.1	69.1	71.0
C-GCN† [39]	68.5	64.4	66.3	–	–	74.6
Transformer − based						
SpanBERT‡ [41]	70.8	70.9	70.8	–	–	78.0
KnowBERT‡ [28]	–	–	70.5	–	–	79.3
MTB [10]	–	–	70.1	–	–	–
LUKE [29]	–	–	72.7	–	–	80.6
RIB$^{*}_{base}$	72.7	68.8	70.7	78.5	78.2	78.3
RIB$^{*}_{large}$ [8]	**77.9**	71.2	74.4	81.8	84.7	83.2
CLERE$_{base}$	74.7	69.8	72.2	**82.6**	80.5	81.5
CLERE$_{large}$	75.3	**74.4**	**74.9**	81.6	**86.4**	**83.9**

¬ Marks re-implemented results from [37]
†Marks re-implemented results from [36]
‡Marks re-implemented results from [13]
* Marks our re-implemented results

is utilized as the default input for downstream tasks or models by researchers. However, PLMs are multi-layer structural models, and the representations of different levels are captured by different layers. Different granularities of feature information are exhibited at different levels. A pivotal point in the fine-tuning task is to obtain the optimal feature information from each level when the downstream tasks differ. Analysis as Fig. 6 in question depicts the self-attention distribution within the partial self-attention layer of PLMs (using BERT-base as an example). The illustration of attention distribution enables a refined scrutiny of the model's allocation of attention to various segments of the input text within each self-attentive layer [42, 43], endowing an intricate insight into the model's underlying mechanisms.

Table 4 Precision, recall, and F1(in %) on RE-TACRED dataset

Models	RE-TACRED		
	Precision	Recall	F1
RNN − based			
PA-LSTM† [36]	79.2	79.5	79.4
GCN − based			
C-GCN† [39]	80.9	79.7	80.3
Transformer − based			
SpanBERT‡ [41]	79.2	79.5	85.3
LUKE [29]	–	–	90.3
RIB$^{*}_{base}$	87.7	88.2	87.9
RIB$^{*}_{large}$ [8]	**91.2**	91.1	**91.1**
CLERE$_{base}$	86.6	90.5	88.5
CLERE$_{large}$	88.2	**94.3**	**91.1**

†Marks re-implemented results from [36]
‡Marks re-implemented results from [13]
* Marks our re-implemented results

The attention of individual tokens to each other is depicted in the diagram, with the thickness of the lines indicating the corresponding attention values as shown in Fig. 6. Notably, a relatively even attention distribution among tokens is observed in the first layer. However, by the second layer, attention becomes notably concentrated on the "[CLS]" token, with subsequent attentional allocation shifting towards the "[SEP]" token by the seventh layer. The most significant attentional focus is observed on three tokens in the final layer, indicating a continual shift in attentional allocation throughout training. This observation highlights the inadequacy of relying solely on the last layer's hidden state as a contextual

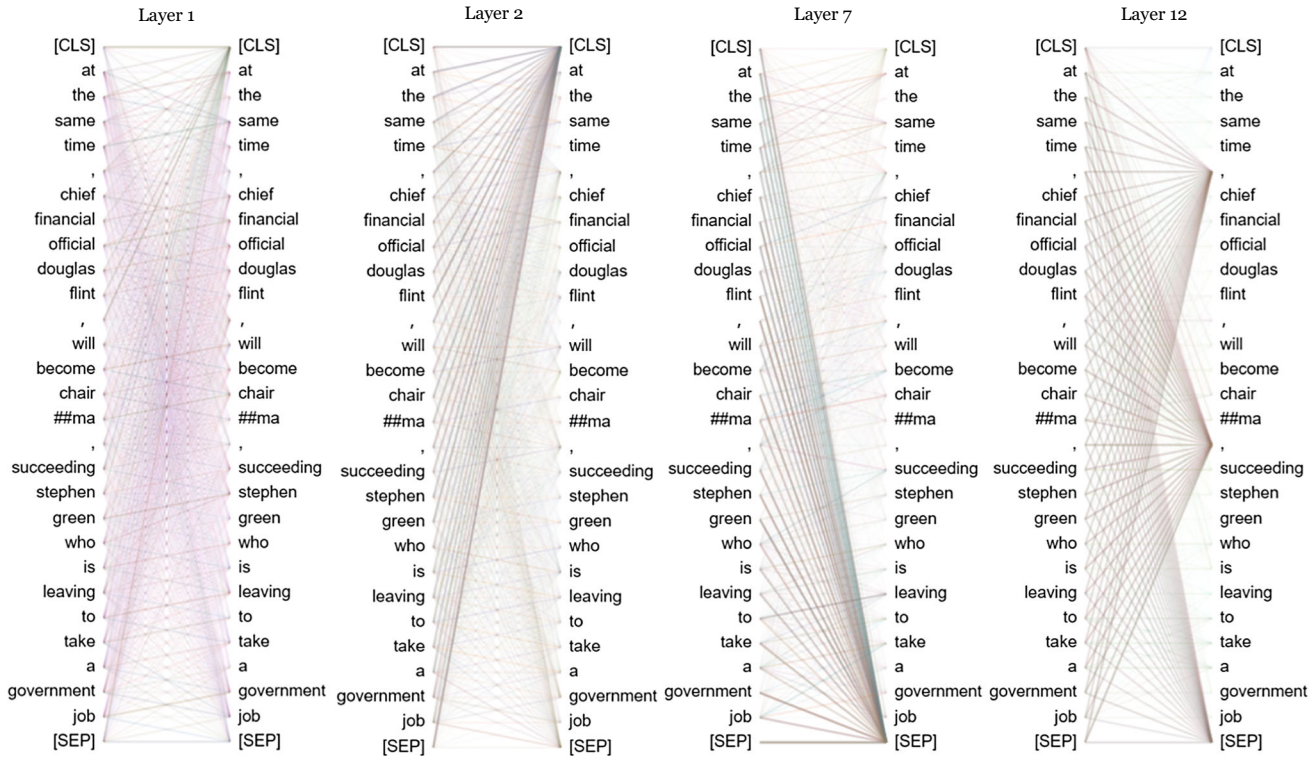

Fig. 6 Self-attention distribution of layers from BERT-base

embedding of the sequence. In our experiments, it is found that considerable attention is placed on non-single tokens in the last four layers of PLMs. This led us to weighted layer pooling, where the contextual representation of the sequence is obtained by combining multiple layers of hidden states. The combination of multiple layers of hidden states as the contextual representation of the sequence is found to be experimentally validated, supporting the validity of our conclusion. Table 5 shows that the weighted layer pooling strategy on the last 4 layers consistently outperformed the other three strategies across all three dataset, demonstrating the effectiveness of this approach for contextual embedding. The ablation experiments also highlighted the importance of carefully selecting pooling strategies to achieve optimal performance in NLP tasks.

Table 5 Ablation experiments on three dataset, using various pooling strategies (F1 scores in %)

Model	TACRED	TACREV	RE-TACRED
+[CLS]	74.2	82.6	90.0
+All layer	73.2	82.1	88.8
+Last 9 layer	74.5	83.1	90.6
+Last 4 layer	74.9	83.9	91.1

Performance on Imbalanced Dataset

Let us take a look at the recall performance of our proposed NLP model on three dataset. Figure 7 provides a clear comparison; using Bert-base as the PLM part, CLERE has better recall performance than most other models. When using Roberta_large, CLERE has the highest recall compared to the control models. This indicates that CLERE can successfully capture a significant portion of the target tokens or categories in the dataset and can effectively identify the target samples in the dataset. This suggests that CLERE is effective in handling unbalanced dataset. The model's robustness is also verified by evaluating its F1 score performance in conjunction with its recall performance.

Sensitivity of Margin

The margin's sensitivity is explored as an essential parameter for triplet loss. It controls the model to differentiate between anchor, positive, and negative examples to make correct judgments during the inference step. From Fig. 8, it can be observed that the effect of the margin on the model is significant. As evidenced by the results obtained from the TACRED (Fig. 8a) and TACREV(Fig. 8b), it can be observed that the classification efficacy of the model diminishes considerably when the margin is increased to 0.35. This

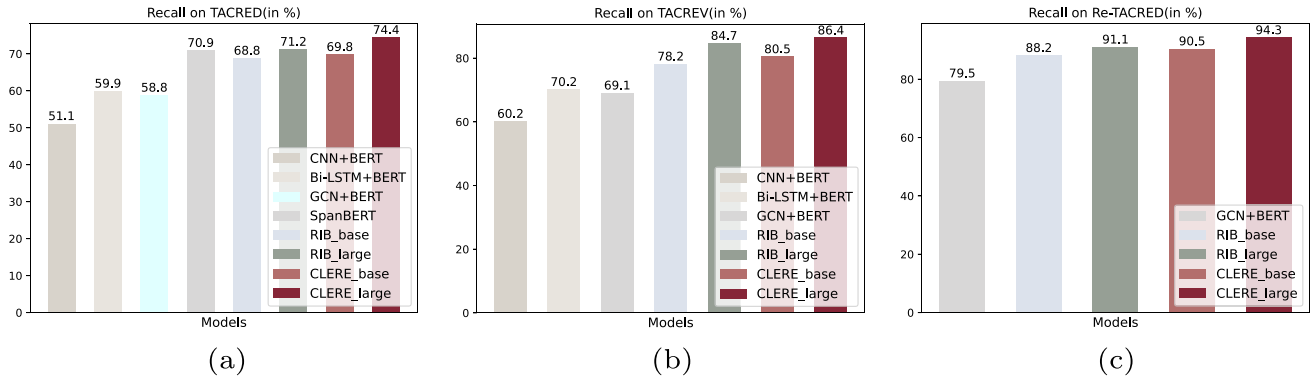

Fig. 7 Recall of the models on three dataset. **a** Shows the recall on the TACRED dataset. **b** Shows the recall results on the TACREV dataset. **c** Shows the recall results on the Re-TACRED dataset

can be attributed to the inherent characteristics of the triplet loss function. The larger margins impede the model's ability to differentiate between positive and negative instances. In contrast, for the RE-TACRED (Fig. 8c), the issue of label imbalance is less pronounced than in the first two dataset, resulting in a diminished sensitivity of the model to variations in margin size. Even when the margin is increased to 0.5, the model maintains a satisfactory level of performance. Based on these observations, it can be concluded that CLERE is well-suited for handling unbalanced dataset and that employing a smaller margin facilitates the model's ability to distinguish between positive and negative instances. For dataset with a more balanced distribution of labels, the model exhibits significantly reduced sensitivity to changes in margin size.

Case Study

Several examples have been listed in Table 6 to provide a clear idea of CLERE. As can be observed from the second sentence, two identical inference results are given by the model, but their inference distances differ, indicating that shortcuts are not taken during the training process. It can also be observed that the model's inference for similar relationships is much closer than that for dissimilar relationships. For instance, in sentence 2, the inference of "no_relation" is closer than that of labels of type "per" and type "org," which is determined by the text semantics of the labels. This provides sufficient evidence for the feasibility of involving semantic information of the labels in the training of the classification task. Moreover, it is found that even if the inferred results are of the same type as the ground truth, such as both "per" types in sentence 1 and "org" types in sentence 3, the model's distance calculation for them is distinguished by more significant differences, illustrating the clarity of CLERE's recognition of label semantics. In summary, the information provided by the data itself has been fully utilized by CLERE.

Error Analysis

Error analysis plays a critical role in identifying model weaknesses, enhancing dataset quality, refining model design, and ultimately improving overall model performance. Table 7 presents a subset of CLERE's inference results on the TACRED dataset, along with predicted outcomes for MTB and RIB. In the inference results for sentence #1, both MTB and RIB yield incorrect predictions as the relationship

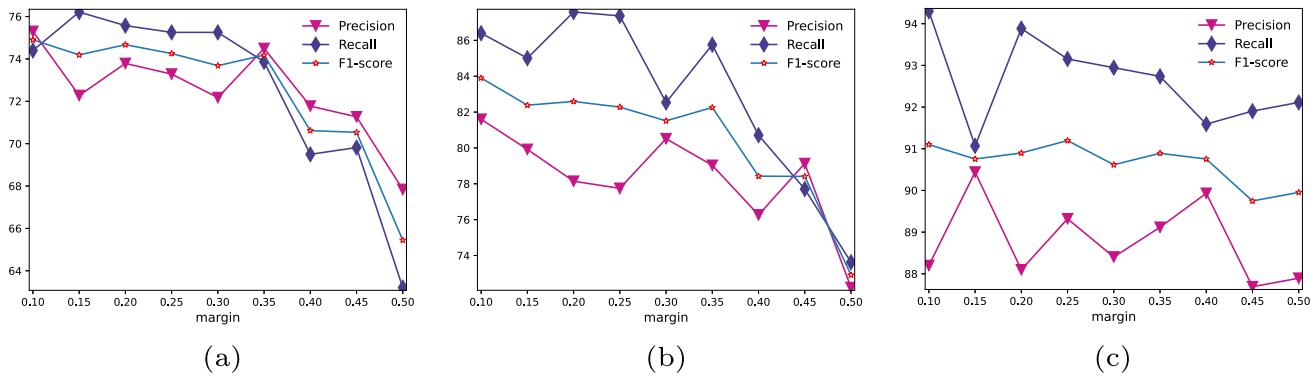

Fig. 8 The sensitivity of margin. Precision, recall, and F1-score on three dataset: **a** On the TACRED dataset. **b** On the TACREV dataset. **c** On the RE-TACRED dataset

Table 6 Case study

Input sentence	Inferenced	Distance	Final result
... Anna Mae Pictou	per:alternate_names	0.184	
as the origin of	per:title	0.336	per:alternate_names
... to Kill Anna Mae.	per:city_of_death	0.473	
per:alternate_names	per:other_family	0.379	
...by the NTSO	no_realtion	0.162	
... with the	no_relation	0.122	no_relation
Sun Moon Lake...Administration	per:children	1.082	
no_relation	org:members	1.205	
...Prachai	org:parents	0.476	
,the founder of	org:member_of	0.509	org:founded_by
Thai Petrochemical Industry	org:website	0.443	
org:founded_by	org:founded_by	0.145	

All the examples are extracted from the test set of TACRED. Blue indicates marked entities, **Magenta** indicates ground truth of the example, Red indicates label selected in the inference step

Table 7 Error analysis, Blue indicates marked entities, **Magenta** indicates ground truth of the example, ✔ indicates that the inference is consistent with ground truth, ✗ indicates that the inference is not consistent with ground truth

Sentence #1 subject_type: PERSON object_type: DURATION sentence length: 18	Salaam, represented by Kunstler at sentencing and in his unsuccessful appeals, got a seven-year term.
ground truth	no_relation
candidate relation	no_relation, per:age, per:date_of_death
inference of MTB	per:age ✗
inference of RIB	per:age ✗
inference of CLERE	no_relation ✔
Sentence #2 subject_type: ORGANIZATION object_type: PERSON sentence length: 61	Suspicions had already fallen on Sheila O'Grady, who is close with David Axelrod and went straight from being former Chicago mayor Richard M. Daley's chief of staff to president of the Illinois Restaurant Association(IRA), as being the person who dug up Herman Cain's personnel records from the National Restaurant Association(NRA).
ground truth	no_relation
candidate relation	org:founded_by, org:founded, per:employee_of
inference of MTB	org:founded_by ✗
inference of RIB	org:founded_by ✗
inference of CLERE	org:founded_by ✗
Sentence #3 subject_type: ORGANIZATION object_type: PERSON sentence length: 34	And strangely enough, Cain's short, three-year tenure at the NRA is evidently the only period in his decades-long career during which he's alleged to have been a sexual predator.
ground truth	org:top_membersemployees
candidate relation	org:top_members/employees, org:founded_by
inference of MTB	org:top_members/employees ✔
inference of RIB	org:top_members/employees ✔
inference of CLERE	org:top_members/employees ✔

between the subject and object is not labeled as "no_relation" by the dataset. Notably, the candidate relations predominantly involve "person" entities, with MTB and RIB inferring the result as "per:age," likely due to the prevalent association between the entities involved. CLERE learns by discerning discrepancies among samples. For instance, if in the training data, the object's type is "DURATION" and the samples are labeled as "no_relation". Consequently, when CLERE encounters analogous situations in the test dataset, the encoding outcome for the test sample will inherently exhibit a diminished semantic distance from "no_relation." In sentence #2, all three methods give incorrect predictions for two possible reasons. Firstly, the subject_type and object_type are "ORGANISATION" and "PERSON," respectively, which have a high similarity within CLERE's candidate relation. Furthermore, from a human perspective, there is a relationship between "Herman Cain" and "National Restaurant Association," which is marked as "no_relation" in the TACRED dataset. Therefore, the imperfections of the dataset could contribute to the model's incorrect predictions. Secondly, based on the statistics from Zhang et al. [36], the proportion of samples with sentence lengths greater than 60 is 3.21%, and this particular sample has a length of 61. Consequently, the ability of the three models to understand long sentences on this dataset is limited, mainly due to the lack of training data for long sentences. Addressing this limitation is of significant practical importance for subsequent improvements. In real-world application scenarios, models often encounter a large number of long sentences, which further emphasizes the need for improved training data coverage in this regard. For sentence #3, the majority of sentence lengths in the TACRED dataset are concentrated between 20 and 42. Consequently, the models achieved their best prediction results when dealing with samples falling within this sentence length range. Overall, based on the error analysis, future efforts should prioritize improving the models' ability to understand longer sentences. CLERE will continue to improve following the work of Zhuang et al. [44] and Wang et al. [45]

Conclusion

In this paper, the RE model is enhanced by improving the pooling strategy and achieving advanced contextual representations. Based on the idea of contrastive learning, the embedding of label semantic information is introduced into the model's learning process, alleviating the distress caused by label imbalance in the dataset. The reasons behind the effect of pooling strategies on contextual embeddings are scrutinized and conducted experiments to investigate their influence on model learning outcomes. The semantic similarity of the labels in the dataset is calculated and find that different labels have significant semantic differences and can be strictly distinguished. Then, the experimental results of the model are analyzed, and the attention distribution of different levels of PLMs is discussed; the ablation experiments are done on kinds of pooling strategies. Furthermore, the impact of the margin on the model's performance is also analyzed. Finally, we demonstrated the proposed method clearly through a case study. We hope that more researchers are willing to explore the role of label semantics.

Acknowledgements This work was supported in part by the National Key Research and Development Program of China (No.2020YFC2003502), the National Natural Science Foundation of China (No.62276038), the Foundation for Innovative Research Groups of Natural Science Foundation of Chongqing (No.cstc2019jcyjcxttX0002), and the Key Cooperation Project of Chongqing Municipal Education Commission (HZ2021008).

Author Contributions All authors contributed to the study conception and design. Zhenyu Zhou contributed to the conception of the study and performed the experiments. Qinghua Zhang and Fan Zhao contributed to the manuscript preparation. Zhenyu Zhou and Fan Zhao performed the experiment analysis and wrote the manuscript. Qinghua Zhang helped perform the analysis with constructive discussions.

Data Availability The datasets during the current study are available in the LDC, Github, and arXiv repositories, https://catalog.ldc.upenn.edu/LDC2018T03, https://github.com/DFKI-NLP/tacrev, arXiv:2104.08398.

Declarations

Conflict of Interest The authors declare no Conflict of interest.

Ethical Approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Nguyen T, Grishman R. Event detection and domain adaptation with convolutional neural networks. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing. 2015. p. 365-71.
2. Lample G, Ballesteros M, Subramanian S, Kawakami K, Dyer C. Neural architectures for named entity recognition. In: Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies. 2016. p. 260-70.
3. Zha E, Zeng D, Lin M, Shen Y. CEPTNER: contrastive learning enhanced prototypical network for two-stage few-shot named entity recognition. Knowl-Based Syst. 2024;295:111730.
4. Chen W, Hong D, Zheng C. Learning knowledge graph embedding with entity descriptions based on LSTM networks. In: 2020 IEEE International Symposium on Product Compliance Engineering-Asia (ISPCE-CN). 2020. p. 1-7.
5. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A, et al. Attention is all you need. In: Advances in neural information processing systems, vol. 30. 2017. p. 6000–10.

6. Devlin J, Chang M, Lee K, Toutanova K. 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. 2019. p. 4171-86.
7. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, et al. RoBERTa: a robustly optimized BERT pretraining approach. 2019. arXiv:1907.11692
8. Zhou W, Chen M. An improved baseline for sentence-level relation extraction. In: Proceedings of the 2nd conference of the Asia-Pacific chapter of the association for computational linguistics and the 12th international joint conference on natural language processing. 2022. p. 161-8.
9. Wang X, Gao T, Zhu Z, Zhang Z, Liu Z, Li J, et al. KEPLER: a unified model for knowledge embedding and pre-trained language representation. Trans Assoc Comput Linguistics. 2021;9:176–94.
10. Baldini Soares L, FitzGerald N, Ling J, Kwiatkowski T. Matching the blanks: distributional similarity for relation learning. In: Proceedings of the 57th annual meeting of the association for computational linguistics. 2019. p. 2895-2905.
11. Wu S, He Y. Enriching pre-trained language model with entity information for relation classification In: Proceedings of the 28th ACM international conference on information and knowledge management. 2019. p. 2361-64.
12. Li Z, Sharaf M, Sitbon L, Du X, Zhou X. CoRE: a context-aware relation extraction method for relation completion. In: 2023 Third International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT). 2023. p. 1-4.
13. Huang J, Li B, Xu J, Chen M. Unified semantic typing with meaningful label inference. In: Proceedings of the 2022 conference of the north american chapter of the association for computational linguistics: human language technologies. 2022. p. 2642-54.
14. Reimers N, Gurevych I. 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). 2019. p. 3980-90.
15. Nayak Y, Majumder N, Goyal P, Poria S. Deep neural approaches to relation triplets extraction: a comprehensive survey. Cogn Comput. 2021;5(13):1215–32.
16. Mondal A, Cambria E, Das D, Hussain A, Bandyopadhyay S. Relation extraction of medical concepts using categorization and sentiment analysis. Cogn Comput. 2018;10:670–85.
17. Peng H, Gao T, Han X, Lin Y, Li P, Liu Z, et al. Learning from context or names? An empirical study on neural relation extraction. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020. p. 3661-72.
18. Hu M, Zhang C, Ma F, Liu C, Wen L, Yu P. Semi-supervised relation extraction via incremental meta self-training. In: Findings of the association for computational linguistics: EMNLP 2021. 2021. p. 487-96.
19. Gao T, Yao H, Chen D. SimCSE: simple contrastive learning of sentence embeddings. In: Proceedings of the 2021 conference on empirical methods in natural language processing. 2021. p. 6894-910.
20. Khosla P, Teterwak P, Wang C, Sarna A, Tian Y, Isola P, et al. Supervised contrastive learning. In: Proceedings of the 34th international conference on neural information processing systems, vol. 33. 2020. p. 18661-73.
21. Nguyen D, Matsuo Y, Ishizuka M. Subtree mining for relation extraction from Wikipedia. In: Human language technologies 2007: the conference of the North American chapter of the association for computational linguistics. 2007. p. 125-28.
22. Liu C, Sun W, Chao W, Che W. Convolution neural network for relation extraction. Advan Data Mining Appl. 2013;8347:231–42.
23. Zeng D, Liu K, Lai S, Zhou G, Zhao J. Relation classification via convolutional deep neural network. In: Proceedings of COLING 2014, the 25th international conference on computational linguistics. 2014. p. 2335-44.
24. Nguyen T, Grishman R. Relation extraction: perspective from convolutional neural networks. In: Proceedings of the 1st workshop on vector space modeling for natural language processing. 2015. p. 39–48.
25. Zhang R, Meng F, Zhou Y, Liu B. Relation classification via recurrent neural network with attention and tensor layers. Big Data Mining Anal. 2018;3(1):234–44.
26. Xu Y, Mou L, Li G, Chen Y, Peng H, Jin Z. Classifying relations via long short term memory networks along shortest dependency paths. In: Proceedings of the 2015 conference on empirical methods in natural language processing. 2015. p. 1785-94.
27. Xu S, Sun S, Zhang Z, Xu F, Liu J. BERT gated multi-window attention network for relation extraction. Neurocomputing. 2022;492:516–29.
28. Peters M, Neumann M, Logan R, Schwartz R, Joshi V, Singh S, et al. Knowledge enhanced contextual word representations. 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). 2019. p. 43-54.
29. Yamada I, Asai A, Shindo H, Takeda H, Matsumoto Y. LUKE: deep contextualized entity representations with entity-aware self-attention. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020. p. 6442-54.
30. Li C, Tian Y. Downstream model design of pre-trained language model for relation extraction task. 2020. arXiv:2004.03786
31. Hadsell R, Chopra S, LeCun Y. Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). 2006. p. 1735-42.
32. Liu J, Liu J, Wang Q, Wang J, Wu W, Xian Y, et al. RankCSE: unsupervised sentence representations learning via learning to rank. In: Proceedings of the 61st annual meeting of the association for computational linguistics. 2023. p. 13785-802.
33. Gunel B, Du J, Conneau A, Stoyanov V. Supervised contrastive learning for pre-trained language model fine-tuning. 2021. arXiv:2011.01403
34. Chen T, Shi H, Tang S, Chen Z, Wu F, Zhuang Y. CIL: contrastive instance learning framework for distantly supervised 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. 2021. p. 6191–200.
35. Zhu X, Meng Q, Ding B, Gu L, Yang Y. Weighted pooling for image recognition of deep convolutional neural networks. Clust Comput. 2019;22(Suppl 4):9371–83.
36. Zhang Y, Zhong V, Chen D, Angeli G, Manning C. Position-aware attention and supervised data improve slot filling. In: Proceedings of the 2017 conference on empirical methods in natural language processing. 2017. p. 33-45.
37. Alt C, Gabryszak A, Hennig L. TACRED revisited: a thorough evaluation of the TACRED relation extraction task. In: Proceedings of the 58th annual meeting of the association for computational linguistics. 2020. p. 1558-69.
38. Stoica G, Platanios E, Poczos B. Re-TACRED: addressing shortcomings of the TACRED dataset. In: Proceedings of the AAAI conference on artificial intelligence, vol. 35. 2021. p. 13843-50.
39. Zhang Y, Qi P, Manning C. Graph convolution over pruned dependency trees improves relation extraction. In: Proceedings of the 2018 conference on empirical methods in natural language processing. 2018. p. 2205-15.

40. Kipf T, Welling M, Manning C. Semi-supervised classification with graph convolutional networks. 2017. arXiv:1609.02907
41. Joshi M, Chen D, Liu Y, Weld D, Zettlemoyer L, et al. SpanBERT: improving pre-training by representing and predicting spans. Trans Assoc Comput Linguistics. 2020;8:64–77.
42. Yamamoto Y, Matsuzaki T. Absolute position embedding learns sinusoid-like waves for attention based on relative position. In: Proceedings of the 2023 conference on empirical methods in natural language processing. 2023. p. 15-28.
43. Klein T, Nabi M. miCSE: mutual information contrastive learning for low-shot sentence embeddings. In: Proceedings of the 61st annual meeting of the association for computational linguistics. 2023. p. 6159-77.
44. Zhuang J, Jing X, Jia X. Mining negative samples on contrastive learning via curricular weighting strategy. Inf Sci. 2024; 668:120534.
45. Wang T, Chen L, Zhu X, Lee Y, Gao J. Weighted contrastive learning with false negative control to help long-tailed product classification. In: Proceedings of the 61st annual meeting of the association for computational linguistics. 2023. p. 574-80.