Improving Continual Few-shot Relation Extraction through Relational Knowledge Distillation and Prototype Augmentation

Zhiheng Zhang[2,*], **Daojian Zeng**[1,2,*,†], **Xue Bai**[3]
[1]Institute of AI and Targeted International Communication, Hunan Normal University, Changsha, China
[2]Hunan Provincial Key Laboratory of Intelligent Computing and Language Information Processing, Hunan Normal University, Changsha, China
[3]Changsha Dworld AI Tech Co.,Ltd., China
{zhangzhiheng,zengdj}@hunnu.edu.cn, baixue@mail.chancein.cn

Abstract

In this paper, we focus on the challenging yet practical problem of Continual Few-shot Relation Extraction (CFRE), which involves extracting relations in the continuous and iterative arrival of new data with only a few labeled examples. The main challenges in CFRE are overfitting due to few-shot learning and catastrophic forgetting caused by continual learning. To address these problems, we propose a novel framework called RK2DA, which seamlessly integrates prototype-based data augmentation and relational knowledge distillation. Specifically, RK2DA generates pseudo data by introducing Gaussian noise to the prototype embeddings and utilizes a novel two-phase multi-teacher relational knowledge distillation method to transfer diverse knowledge from different embedding spaces. Experimental results on the FewRel and TACRED datasets demonstrate that our method outperforms the state-of-the-art baselines.

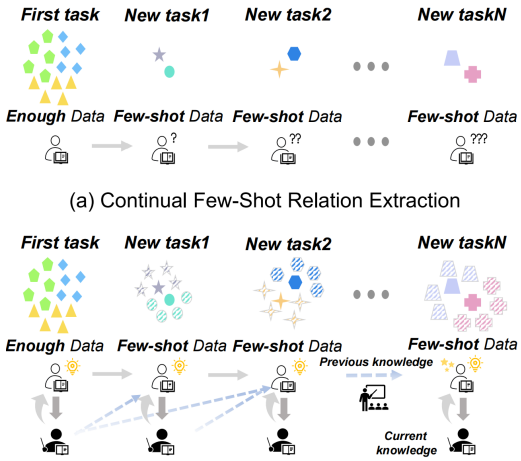
Keywords: continual few-shot learning, relational knowledge distillation, prototype data augmentation

1. Introduction

Relation Extraction (RE) is a crucial component of NLP which focuses on automatically identifying the relation between two named entities mentioned within a sentence for various downstream applications (Wang et al., 2015; Yu et al., 2017). However, traditional RE methods (Zeng et al., 2014; Baldini Soares et al., 2019) exhibit limitations in handling the rapid emergence of novel relations in real-world scenarios, since they perform once-and-for-all training on a predefined and fixed set.

To adapt to this situation, Continual Relation Extraction (CRE) was introduced (Wang et al., 2019). Compared with traditional RE, CRE demands that models to retain a stable understanding of old relations and incrementally learn new tasks. A straightforward solution is to store all previous data and combine it with new data for model retraining. Unfortunately, it is impracticable due to limitations in storage and computing resources. Thus, CRE suffers from catastrophic forgetting (McCloskey and Cohen, 1989; French, 1999), where the model tends to forget previous knowledge and the embedding space will gradually be destroyed.

There are three primary approaches to address this problem: regularization-based methods, dynamic architecture methods, and memory-based methods. Among them, memory-based methods have demonstrated superior performance in NLP scenarios (Wang et al., 2019; Han et al., 2020; Cui et al., 2021; Zhao et al., 2022; Hu et al., 2022; Zhang et al., 2022). These methods store several key examples from previous tasks in a memory module and utilize them for subsequent task learning. However, despite their effectiveness, they all rely on extensive annotated data for new relations. This reliance presents challenges in real-life scenarios where acquiring sufficient labeled data for continuously emerged relations is often costly and impracticable. As a result, this issue gives rise to a long-tail distribution of relations in the real world, where novel relations are few-shot with a limited



(a) Continual Few-Shot Relation Extraction

(b) Continual Few-Shot Relation Extraction + RK2DA

Figure 1: Comparisons of conventional CFRE and our method RK2DA. Conventional CFRE continuously learns relations from few-shot data stream, which suffers from catastrophic forgetting and overfitting. RK2DA generates pseudo data and transfers various knowledge to alleviate these problems.

* indicates equal contribution
† Corresponding author

LREC-COLING 2024, pages 8756–8767
20-25 May, 2024.

number of samples. Therefore, Continual Few-shot Relation Extraction (CFRE) was introduced (Qin and Joty, 2022). As shown in fig. 1(a), CFRE obtains knowledge of novel relations from a continuous few-shot data stream. Therefore, CRFE not only faces the challenge of catastrophic forgetting but also encounters the overfitting conundrum that arises from few-shot examples.

To tackle above problems, some attempts have been made, such as the method proposed by Qin and Joty, which is based on embedding space regularization and data augmentation (ERDA). ERDA imposes constraints on the embedding space and uses a self-supervised method to retrieve sentences with the same entity or high similarity scores for data augmentation. However, this method struggles to find sentences with correct relations and ensure balanced data volume for each relation. Moreover, ERDA does not utilize history knowledge obtained from previous tasks.

In fact, humans can continually draw inferences from a handful of examples through repetition of history knowledge and reconsolidation exercises (Tononi and Cirelli, 2006; Boyce et al., 2016). Based on this observation, we propose a novel method, **RK2DA**, which seamlessly integrates **R**elational **K**nowledge **D**istillation and Prototype **D**ata **A**ugmentation (Park et al., 2019; Zhu et al., 2021; Thi et al., 2022) into leraning framework.

As shown in fig. 1(b), RK2DA utilizes a combination of pseudo data (represented by shaded data points) and various knowledge (both previous and current knowledge) for training. Specifically, after rapidly adapting current knowledge, we introduce Gaussian noise to the relation prototypes for data augmentation. This simpler method efficiently generates diverse data to form a balanced fine-tuning. Training with these sufficient pseudo data enables model to reconsolidate both the previous and current knowledge, thus alleviating the overfitting problem. Afterwards, considering that different teacher imparts different knowledge, we propose a novel two-phase multi-teacher relational knowledge distillation approach. It transfers various knowledge by comparing relation prototypes between new and old embedding spaces in two phases. Such transfer of correct knowledge ensures alignment and uniformity between the data distributions of different tasks, rectifies incorrect knowledge and mitigates the catastrophic forgetting. Finally, we use a simple reconsolidation module to consolidate the knowledge acquired from the current task. Our contributions in this paper are summarized as follows:

- We propose RK2DA, a novel method that seamlessly integrates relational knowledge distillation and prototype data augmentation into learning framework, to fully alleviate catastrophic forgetting and overfitting in CFRE.
- The modules of RK2DA effectively utilize various knowledge and augmented pseudo data, thereby ensuring a stable comprehension of previous knowledge while learning new tasks.
- Extensive experiments demonstrate that our RK2DA outperforms the state-of-the-art (SOTA) models on two benchmark datasets, FewRel and TACRED.

2. Related Work

2.1. Continual Relation Extraction

Traditional RE methods mainly include supervised methods (Liu et al., 2013; Zeng et al., 2014), semi-supervised methods (Chen et al., 2006; Hu et al., 2021), and distant supervised methods (Yao et al., 2011; Zeng et al., 2015; Han et al., 2018). However, these methods all perform once-and-for-all training on a predefined static relation set, without considering the continuous emergence of new relations in the real world. Hence, CRE aims to extract relations from a continuous data stream.

The main challenge in CRE is catastrophic forgetting (McCloskey and Cohen, 1989; French, 1999). Current methods used to alleviate this problem can be divided into three categories. (i) Regularization-based methods impose constraints on the update of parameters (Li and Hoiem, 2018; Kirkpatrick et al., 2017; Ritter et al., 2018). (ii) Dynamic architecture methods change models' architectural properties upon new data by dynamically accommodating new neural resources (Chen et al., 2016; Fernando et al., 2017; Mallya et al., 2018). (iii) Memory-based methods explicitly retrain the models on a limited subset of stored samples (Han et al., 2020; Cui et al., 2021; Thi et al., 2022). Among these methods, memory-based methods have been proven to be the most promising in NLP tasks (Wang et al., 2019). Inspired by the success of memory-based methods in CRE, we continue to utilize the memory-based approach. Additionally, Wang et al. has highlighted the issue of data imbalance between new and old relations leads to performance degradation in previous work (Han et al., 2020; Cui et al., 2021). Therefore, we generate balanced training set to ensure each relation has an equal number for fine-tuning.

2.2. Continual Few-Shot Relation Extraction

In real-world scenarios, obtaining a substantial amount of annotated data is expensive and impracticable. Thus, the concept of CFRE was introduced. In contrast to Few-Shot Relation Extraction (FSRE) (Gao et al., 2019; Yang et al., 2021; Ren et al., 2023), where models learn from a few-shot but still fixed set, CFRE presents a greater challenge as it

strives to achieve continual learning and few-shot learning simultaneously. In CFRE, the model needs to continually learn relational patterns from a sequence of few-shot tasks, which suffers from both the problems of catastrophic forgetting and overfitting. Qin and Joty were the first to explore this field and highlighted that SOTA efficient methods for CRE may not be applicable under the continual few-shot setting. They proposed a novel method called ERDA, which enforces additional constraints on the relational embeddings and adds relevant data in a self-supervised manner. While ERDA improved CFRE performance, it overlooks the utilization of knowledge from previous tasks and encounters difficulties in identifying correct data. Hence, we integrate relational knowledge distillation and prototype data augmentation to address these limitations.

2.3. Knowledge Distillation

Knowledge distillation has gained significant attention as a method for compressing and accelerating models. It involves training smaller student models by learning from larger teacher models. Since knowledge distillation enables the transfer of expertise from different models, it has been introduced into the field of continual learning. For example, Learning without Forgetting (LwF) (Li and Hoiem, 2018), Replay-through-Feedback (RtF) (De Lange et al., 2022) and Lifelong Language Knowledge Distillation (L2KD) (Chuang et al., 2020). Some prior studies have also used knowledge distillation in CRE to transfer previous knowledge and maintain the stability of the embedding space (Zhao et al., 2022; Thi et al., 2022). Dong et al. firstly applied relational knowledge distillation to the problem of continual few-shot learning in computer vision, which made great success. However, previous work all focus on transferring knowledge solely from the last model. Considering that different teachers impart different knowledge, we propose a novel two-phase multi-teacher framework to transfer various knowledge from different embedding spaces.

3. Methodology

3.1. Task Formulation

CFRE involves learning from a sequence of n tasks $\mathbb{T} = \left(T^1, \ldots, \mathcal{T}^n\right)$. Each task $\mathcal{T}^k$ has its own training set D^k_{train}, test set D^k_{test} and relation set R^k. Every dataset D contains several samples $\{(x_i, y_i)\}_{i=1}^{|D|}$, where (x_i, y_i) represents the relation of an entity pair in sentence x_i is $y_i \in R^k$. To address the issue of catastrophic forgetting, we follow the memory-based methods setting, utilizing a memory module $\mathcal{M} = \{\mathcal{M}^1, \mathcal{M}^2, ...\}$. The memory $\mathcal{M}$ stores key samples from previous tasks. $\widetilde{D}^k_{\text{train}}$ denotes the augmented training set, which is expanded through the scale set $\mathcal{S}$. $\hat{(\cdot)}^k$ represents the union corresponding to $(\cdot)$ at stage k, such as the memory module $\hat{M}^k = \cup_{i=1}^{k} M^i$.

The principal distinction between CFRE and CRE resides in the assumptions regarding available training data. CFRE assumes sufficient training data for $\mathcal{T}^1$, while the subsequent tasks are few-shot, with only a limited number of labeled instances. Assuming that the number of relations in each few-shot task is N and the number of samples for each relation is K, this setup is termed **N-way K-shot** continual learning. The problem setup of CFRE aligns with the real scenarios, where there is generally sufficient data for existing tasks, but only a handful of labeled data for new tasks.

Algorithm 1 Training process at time step k

Input: the training set D^k_{train} and the relation set R^k of the current task $\mathcal{T}^k$, the current memory $\hat{\mathcal{M}}^{k-1}$ and the known relation set $\hat{R}^{k-1}$, all history relation scale set $\hat{\mathcal{S}}^{k-1}$, distillation frequency N.
Output: f^k_θ, M^k, $\hat{R}^k$,$\hat{\mathcal{S}}^k$

Initialize the embeddings r_i for R^k,$\widetilde{D}^k_{train} = \emptyset$
for $i = 1, \ldots, epoch_1$ **do**
  Update θ with $\nabla\mathcal{L}_{FA}$
end for
Store key samples from D^k_{train} in $\mathcal{M}^k$
Compute the scale of every relation $r_i \in R^k \cup R^1$ to store in $\mathcal{S}^k$
Update $\hat{R}^k, \hat{\mathcal{M}}^k, \hat{\mathcal{S}}^k$
for $r_i \in \hat{R}^k$ **do**
  Compute the scale s_{r_i} of r_i through $\hat{\mathcal{S}}^k$
  Generate expanded $\widetilde{D}^k_{train,r_i}$ for r_i
  $\widetilde{D}^k_{train} = \widetilde{D}^k_{train} \cup \widetilde{D}^k_{train,r_i}$
end for
for $i = 1, \ldots, epoch_2$ **do**
  for $j = 2, \ldots, iter$ **do**
    Update θ with $\nabla\mathcal{L}_{FA}$
    if j==N **then**
      Update θ with knowledge Distillation $\nabla\mathcal{L}_{RKD_1}$
      Update θ with knowledge Distillation $\nabla\mathcal{L}_{RKD_2}$
    end if
  end for
end for
for $i = 1, \ldots, epoch_3$ **do**
  Update θ with $\nabla\mathcal{L}_{FA}$
  Update θ with knowledge Distillation $\nabla\mathcal{L}_{RKD_2}$
end for

3.2. Framework Overview

The framework of RK2DA is shown in fig. 2 and detailed learning procedures are illustrated in algorithm 1 which consists of three major steps: (i) **Fast Adaption (FA)** (line $2 \sim 4$, fig. 2(a)) The encoder's parameters are trained on the $\mathcal{M}^1 \cup D^k_{train}$ (D^k_{train}

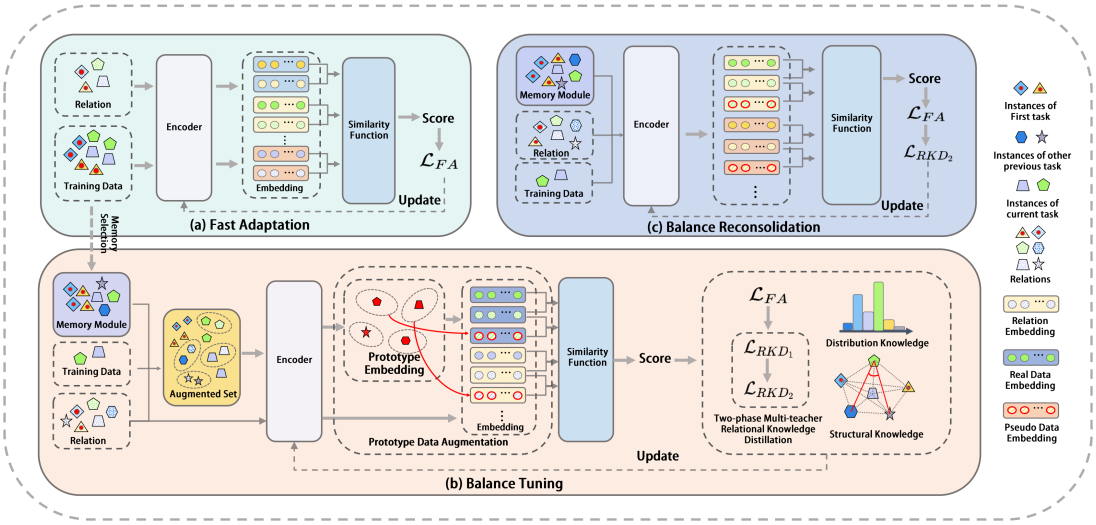


Figure 2: The overall framework of our proposed RK2DA.

for $\mathcal{T}^1$) with $\mathcal{L}_{\mathcal{FA}}$ to obtain knowledge in and between $\mathcal{T}^k$ and $\mathcal{T}^1$ (ii) **Balance Tuning (BT)** (line $5 \sim 21$) After the FA, for each relation $r_i \in R^k$, we use the k-means algorithm to select key instances from D^k_{train} to store in memory (line $5 \sim 6$). Unlike traditional CRE, we only store one instance for each relation in few-shot tasks. Meanwhile, we also record the scale s^k_r of relation in this time step. Then, we generate an augmented $\widetilde{D}^k_{\text{train}}$ where each relation has an equal number of data for balanced fine-tuning (line $8 \sim 12$). Afterwards, we generate pseudo data based on prototype data augmentation and use two-phase multi-teacher relational knowledge distillation to transfer knowledge from both previous and current tasks in the embedding space (line $13 \sim 21$). (v) **Balance Reconsolidation (BR)** (line $22 \sim 25$, fig. 2(c)) In BT, we primarily focus on the restoration of the embedding space disrupted by FA. This increases complexity of the current task learning. Hence, we carry out reconsolidation to enhance learning performance of current task in the restored space. For a predicted instance x_i, we calculate its cosine similarity to all relations' prototype embedding, and select the highest relation y^*_i as i's predicted relation. Next, we'll first introduce basic encoder network and relational knowledge distillation. Afterwards, we will provide a detailed description of each module.

3.3. The Encoder Network

The siamese encoder (f_θ) aims to extract generic relation related features from input. The input can be a labeled sentence or the name of a relation. We use same encoder **Bi-LSTM** as (Han et al., 2020; Qin and Joty, 2022) to conduct a comprehensive comparison with SOTA models. In the classical CRE and CFRE methods (Wang et al., 2019; Han et al., 2020; Qin and Joty, 2022), **Bi-LSTM** is widely used. It takes GloVe embeddings (Pennington et al., 2014) of the words in a given input and produces a vector representation through a Bi-LSTM (Hochreiter and Schmidhuber, 1997).

3.4. Relational Knowledge Distillation

As Relational Knowledge Distillation (RKD) constitutes the core component of the subsequent training stage section 3.7, we will introduce the details of this technology in this section. RKD aims to transfer structural knowledge (e.g., angle-wise or distance-wise relations) from the teacher's output presentation. However, embeddings of text data change during training. If we simply restrict the distance, it will reduce the flexibility of the model and make the embedding space too stable to be compatible with subsequent knowledge. Thus, we replace it with conventional KL divergence loss.

KL divergence loss (Zhao et al., 2022) We use the similarity metric between relations as distribution knowledge to maintain consistency in the distribution of old relations. Specifically, we will calculate the prototype μ^k_r of each relation. Then, the cosine similarity between the classes is calculated to represent the distribution knowledge:

$$a_{r_i,r_j} = \frac{\mu^T_{r_i}\mu_{r_j}}{\|\mu_{r_i}\| \left\|\mu_{r_j}\right\|} \tag{1}$$

where a_{r_i,r_j} is the cosine similarity between prototype r_i and r_j. The prototype is calculated by:

$$\mu^k_r = \frac{1}{|D_r|+1} \cdot \left(\sum_{x_i \in D_r} f_\theta(x_i) + \mathbf{r} \right) \tag{2}$$

Then, we use KL divergence to make the encoder retain the distribution knowledge of the old tasks:

$$\mathcal{L}_{KL}(R, D; \theta^t, \theta^s) = \sum_i KL\left(P_i; \theta^t || Q_i; \theta^s\right) \tag{3}$$

where R is relation set and D is dataset for calculating loss, P_i is the metric distribution of prototype before training, and $p_{r_i r_j} = \frac{\exp(a_{r_i r_j}/\tau)}{\sum_{r_j} \exp(a_{r_i r_j}/\tau)}$. Similarly, Q_i is the metric distribution during training, and $q_{r_i r_j} = \frac{\exp(\tilde{a}_{r_i r_j}/\tau)}{\sum_{r_j} \exp(\tilde{a}_{r_i r_j}/\tau)}$. $\tilde{a}$ is the cosine similarity of temporary prototypes during training.

Angle-wise distillation loss (Park et al., 2019) We use angle-wise distillation loss to transfer the relationship of training embeddings by penalizing angular differences. Given a triplet of embeddings, an angle-wise relational potential measures the angle formed by the three embeddings:

$$\psi_{\mathrm{A}}(x_i, x_j, x_k) = \cos \angle x_i x_j x_k = \langle \mathbf{e}^{x_i x_j}, \mathbf{e}^{x_k x_j} \rangle$$
$$\text{where} \quad \mathbf{e}^{ab} = \frac{f_\theta(a) - f_\theta(b)}{\|f_\theta(a) - f_\theta(b)\|_2}. \tag{4}$$

Using the angle-wise potentials measured in the previous tasks and current task embedding spaces, an angle-wise distillation loss is defined as:

$$\mathcal{L}_A(R, D; \theta^t, \theta^s) = \sum_{(x_i, x_j, x_k) \in \mathcal{X}^3} l_\delta(\psi_{\mathrm{A}}(x_i, x_j, x_k; \theta^t), \psi_{\mathrm{A}}(s_i, s_j, s_k; \theta^s)) \tag{5}$$

where l_δ is the Huber loss. The final RKD loss function consists of two parts:

$$\mathcal{L}_{\mathrm{RKD}}(R, D; \theta^t, \theta^s) = \lambda_{KL} \cdot \mathcal{L}_{KL} + \lambda_A \cdot \mathcal{L}_A \tag{6}$$

Our focus is on a more challenging CFRE problem, where what knowledge to transfer and how to effectively transfer knowledge are equally important. Previous knowledge distillation methods used in continual learning have mainly focused on single-phase single-teacher methods, which only emphasize knowledge from the last task. These methods overlook the catastrophic forgetting of previous task which results in the transfer of incorrect knowledge. Considering that different teacher impart different knowledge, we propose a two-phase multi-teacher learning method for transferring correct knowledge from different embedding spaces. We will provide a detailed introduction to this method in Balance Tuning (section 3.7).

3.5. Fast Adaption for New Task

Since the relations in $\mathcal{R}^k$ does not appear before, the model is initially fine-tuned to obtain knowledge from new task. Unlike traditional approaches in CRE and CFRE that only use the current training set for fine-tuning, we train the model on the $D^k_{train} \cup \mathcal{M}^1(D^1_{train}$ for the $\mathcal{T}^1)$, hoping to obtain the knowledge in and between the new task and the base task. There is no class imbalance issue since we store the same number as the new task for the first task. Then, we optimize the parameters (θ) by minimizing a loss $\mathcal{L}_{FA}$ that consists of a cross entropy loss, a multi-margin loss and a pairwise margin loss. The cross entropy loss $\mathcal{L}_{CE}$ is used for relation classification as follows:

$$\sum_{(x_i, y_i) \in D} \sum_{j=1}^{|R|} \delta_{y_i, r_j} \times \log \frac{\exp\left(g\left(f_\theta(x_i), \mu^k_{r_j}\right)\right)}{\sum_{l=1}^{|R|} \exp\left(g\left(f_\theta(x_i), \mu^k_{r_l}\right)\right)} \tag{7}$$

$k, g(,)$ is a function used to measure similarity between two vectors (e.g., cosine similarity), and $\delta_{a,b}$ is the Kronecker delta function. Additionally, we use two same margin-based losses as ERDA to increase similarity score gap between the correct label and the wrong label. The first one is a multi-margin loss, which is defined as:

$$\mathcal{L}_{\mathrm{mm}} = \sum_{(x_i, y_i) \in D} \sum_{j=1, j \neq t_i}^{R} \max(0, m_1 - g\left(f_\theta(x_i), \mu^k_{r_{t_i}}\right) + g\left(f_\theta(x_i), \mu^k_{r_j}\right)) \tag{8}$$

where t_i denotes the correct relation index within $\hat{R}^k$, such that $r_{t_i} = y_i$, and m_1 represents a specified margin value. The $\mathcal{L}_{mm}$ loss is designed to promote intra-compactness and simultaneously enlarge inter-class distances. The second one is a pairwise margin loss:

$\mathcal{L}_{pm}$:

$$\mathcal{L}_{\mathrm{pm}} = \sum_{(x_i, y_i) \in D} \max(0, m_2 - g\left(f_\theta(x_i), \mu^k_{r_{t_i}}\right) + g\left(f_\theta(x_i), \mu^k_{s_i}\right)) \tag{9}$$

where m_2 is the margin for $\mathcal{L}_{\mathrm{pm}}$ and $s_i = \arg\max_s g\left(f_\theta(x_i), \mu^k_{r_s}\right)$ s.t. $s \neq t_i$, the closest wrong label. The $\mathcal{L}_{\mathrm{pm}}$ try to increase the similarity score gap of the correct label and the closest wrong label. The total loss for FA on $\mathcal{T}^k$ is defined as:

$$\mathcal{L}_{FA} = \lambda_{\mathrm{ce}} \mathcal{L}_{\mathrm{ce}} + \lambda_{\mathrm{mm}} \mathcal{L}_{\mathrm{mm}} + \lambda_{\mathrm{pm}} \mathcal{L}_{\mathrm{pm}} \tag{10}$$

where $\lambda_{\mathrm{ce}}, \lambda_{\mathrm{mm}}$ and λ_{pm} are the relative weights of the component losses, respectively.

3.6. Memory Selection and Data Augmentation

Since Memory selection and data augmentation are important in BT, we introduce them in this section.

For each relation r in R^k, we apply the K-means algorithm to select typical sample into M^k. Specifically, we obtain the embeddings of r's samples from the encoder. Then we calculate the centroid feature by averaging the embeddings of each cluster. Afterwards, we choose the sample per new

relation closest to the centroid and store it in memory $\mathcal{M}^k$ for few-shot tasks. Since the relations in the base task are common, we store K samples per relation, corresponding to the value of K-shot.

Additionally, we record the scale S_r^k in base task and current task for data augmentation in BT:

$$s_r^k = \sqrt{\frac{1}{2}({s_r^{k_*}}^2 + \frac{\mathrm{Tr}\left(\sum_i^k\right)}{N})} \tag{11}$$

where N is the dimension of the embedding space. $\sum_{ri}^k$ is the covariance matrix for the features from relation r at stage k. k^* represents the stage at which the relation r occurs, and the Tr is the trace of a matrix.

Due to privacy concerns and CFRE setting, we cannot store all the samples for few-shot tasks. So, we use prototype augmentation based on the data distribution in the embedding space to reconsolidate both previous and current knowledge.

RK2DA generates $\widetilde{D}^k_{train,r_i}$ for each relation $r_i \in \hat{R}^k$. Each relation has a total of n samples. Specifically, we store n_1 real samples and n_2 prototypes that need to be augmented, where $n_1 + n_2 = n$. When storing real samples, we store all n_1 samples from D^k_{train,r_i}, where $r_i \in R^k$,. For other former relations, we store their memory samples. When storing prototypes, instead of using the current prototype, we store memory samples and relations for n_2 times. During training, when sampling these prototypes, we obtain the prototype with the encoder. This approach allows the model to become more flexible and adaptable as training progresses.

We also store the scale for each relation to generate pseudo data $\widetilde{D}^k_{train}$. In previous step, we have computed $r_i \in R^k \cup R^1$. But for $\cup_{i=2}^{k-1} R^k$, we cannot obtain the true scale value of its distribution through encoder, as we only have one sample per class. So we use the scale values of the base tasks at different stages to estimate the scale of the relations $r \in \cup_{i=2}^{k-1} R^i$ in current training stage as follows:

$$s_{r_j}^k = \sqrt{\frac{1}{2}\left({s_{r_j}^{k_*}}^2 + \left(\frac{1}{|R^1|}\sum_{r_j \in R^1} s_{r_i}^{k^*} \cdot \frac{s_{r_j}^k}{s_{r_j}^{k^*}}\right)^2\right)} \tag{12}$$

We consider various relations with different scales instead of relying on the average scale (Zhu et al., 2021; Thi et al., 2022). As Ren et al. has proven that introducing granularity information is helpful for improving RE performance. So in the BT phase, the relationship r will be augmented with different scales as:

$$f_r^k = \mu_r^k + \epsilon * s_r^k \tag{13}$$

where $\epsilon \sim N(0,1)$ is the Gaussian noise, and the μ_r^k is the expectation feature of r computed by eq. (2) D_r is the Dataset of relation r, D^k_{train} for $r \in R^k$ and M_r for $r \in \cup_{i=1}^{k-1} R^i$.

3.7. Balance Tuning

The goal of BT is to restore the embedding space disrupted by FA and learn new relations while ensuring a stable understanding of previous tasks. in CRE, memory replay with M is often used. However, in CFRE, only one instance of the previous few-shot relations is stored in memory. Using only these limited data for memory replay can lead to severe overfitting and data imbalance problems. Therefore, we fine-tune the model with the expanded dataset $\widetilde{D}^k_{\text{train}}$ in BT. $\widetilde{D}^k_{\text{train}}$ ensures that all relations have an equal number of different instances.

We use the augmented dataset as the training set. When sampling data needs to be augmented, we utilize eq. (13) to obtain its pseudo embedding. When sampling real data, we add a small Gaussian noise to its embedding to prevent overfitting caused by multiple replays.

During training process, we use fundamental $\mathcal{L}_{FA}$ function to acquire knowledge from both new and old tasks. Furthermore, as we consistently achieve a relatively stable embedding space for both the current and previous tasks after each training phase, these embedding spaces encompass various knowledge. Thus, we propose a two-phase multi-teacher RKD approach to effectively acquire knowledge from different tasks at different stages.

Specifically, after updating $\mathcal{L}_{FA}$, we utilize $\mathcal{L}_{\mathrm{RKD}}$ to obtain knowledge from both the old and new task embedding spaces. Our two-phase approach consists of two parts. In the first phase, for the old task, we extract information from the embedding space of the old tasks trained in the previous stages $1, 2, \ldots, k-1$ to obtain knowledge of the old tasks during training. Instead of using all sample pairs in the memory M for knowledge distillation, we utilize the prototypes of each relation, which provides higher flexibility for learning future tasks. Additionally, we emphasizes prototype knowledge of each task to obtain task-level knowledge:

$$\begin{aligned}\mathcal{L}_{RKD_1} = &\sum_{i=1}^{k-1} \lambda_{1,i} \mathcal{L}_{\mathrm{RKD}}(\hat{R}^i, \hat{M}^i; \theta^i, \theta^{now}) \\ &+ \sum_{j=1}^{k} \lambda_{2,j} \mathcal{L}_{\mathrm{RKD}}(R^j, M^j; \theta^j, \theta^{now})\end{aligned} \tag{14}$$

In the second phase, since we have recovered the knowledge of previous tasks, we acquire the current task $\mathcal{T}^k$'s knowledge to better match with the previous knowledge:

$$\mathcal{L}_{RKD_2} = \mathcal{L}_{\mathrm{RKD}}(R^k, M^k; \theta^k, \theta^{now}) \tag{15}$$

To ensure the flexibility of our model without overfitting to the old tasks, we perform knowledge distillation at a fixed frequency.

3.8. Balance Reconsolidation

Han et al. firstly proposed using reconsolidation module to reconsolidate the memory of the relations between old and new tasks. They fine-tune the model on the memory samples together with the current training set. However, Wang et al. has shown that this module introduces the problem of distribution imbalance. In our BT, we are primarily concerned with the restoration of the embedding space with multiple rounds of knowledge distillation. This increases the complexity of the learning process for the current task. Hence, we execute balance reconsolidation to reconsolidate the knowledge of current task in the restored space.

Specifically, we use $\cup_{i=1}^{k-1} M^i$ and D_{train}^k as training sets and utilize $\mathcal{L}_{FA}$ for reconsolidation. However, considering that the issue of data imbalance, we add the same number of samples from the memory module as the D_{train}^k and introduce small Gaussian noise during replay. After each training round, the knowledge distillation helps maintain the stability of the embedding space.

4. Experiments

4.1. Datasets

Our experiments are conducted on two benchmark datasets in the experiment:

FewRel (Han et al., 2018) is a RE dataset that contains 80 relations, each with 700 instances. For the convenient comparison with previous work, we follow the experimental settings in ERDA(Qin and Joty, 2022). We randomly split the relations into 8 tasks with 10 relations per task and we sample 100 samples for relations in $\mathcal{T}^1$ to have enough data. Other tasks $\mathcal{T}^2 \dots \mathcal{T}^8$ are few-shot.

TACRED (Zhang et al., 2017) is a large-scale RE dataset containing 42 relations, we filter out the special relation "n/a" . Similar to FewRel, we split the remaining 41 relations into 8 tasks and randomly sample examples. Except for the first task that contains 6 relations with 100 examples per relation, all other few-shot tasks have 5 relations.

4.2. Evaluation Metric

Following Qin and Joty, the model will be evaluated on the testsets $\hat{D}_{\text{test}}^k = \cup_{i=1}^k D_{\text{test}}^i$ of all seen relations by average accuracy after training on each task. This metric reflects whether the model can alleviate catastrophic forgetting and overfitting while acquiring novel knowledge well with limited data.

Method	Task index							
	1	2	3	4	5	6	7	8
SeqRun	92.78	52.11	30.08	24.33	19.83	16.90	14.36	12.34
Joint Train	92.78	76.29	69.39	64.75	60.45	57.64	52.80	50.03
EMAR	85.20	62.02	52.45	48.95	46.77	44.33	40.75	39.04
ERDA	92.57	79.17	70.43	65.01	61.06	57.54	54.88	53.23
RK2DA	**93.78**	**83.05**	**74.67**	**69.52**	**64.83**	**60.71**	**57.56**	**54.58**

Table 1: Accuracy (%) of different methods at every time step on **FewRel** benchmark for 10-way 5-shot CFRE.

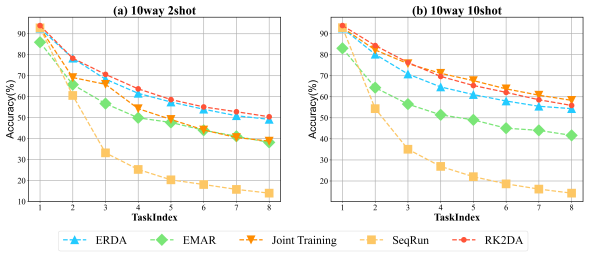


Figure 3: Comparison results at each time step on **FewRel** benchmark for 10-way 2-shot and 10-shot settings.

4.3. Baselines

We compare our approach with the following baselines: (i) **SeqRun** fine-tunes the model only on the training data of the new tasks without using any memory data. It faces serious catastrophic forgetting and serves as a lower bound. (ii) **Joint Training** stores all previous samples in the memory and trains the model on all data for each new task. It serves as an upper bound in CRE. (iii) **EMAR** (Han et al., 2020) adopts memory activation and reconsolidation to alleviate catastrophic forgetting, which is a SOTA method of CRE. (iv) **ERDA** (Qin and Joty, 2022) is SOTA method on CFRE, it uses embedding space regularization and data augmentation to alleviate catastrophic forgetting and overfitting.

4.4. Experiments Settings

Since different task orders have an impact on the model's performance, we set the same random seed as in (Qin and Joty, 2022) to ensure same task order. For other settings, such as hidden embedding dimension and pre-trained input embeddings, we follow the settings in (Qin and Joty, 2022). As $\mathcal{L}_{FA}$ is referenced from ERDA, the relevant parameters such as λ_{ce}, λ_{mm}, λ_{pm}, and learning rate are also consistent with ERDA. For the RKD loss,

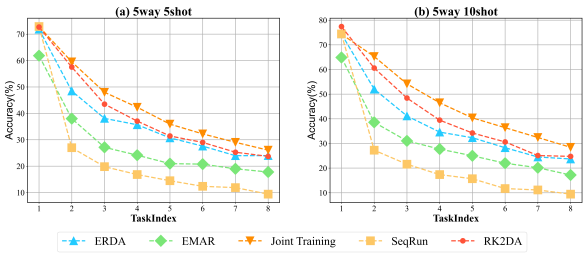


Figure 4: Comparison results at each time step on **TACRED** benchmark for 5-way 5-shot and 10-shot settings.

we set $\lambda_{KL} = 0.5, \lambda_A = 1.0$ (Park et al., 2019). we set $\lambda_{1,k-1} = 0.5$, $\lambda_{1,k-1} = 0.15$, $\lambda_{2,k} = 1, \lambda_{1,k-1} = 0.15$, and the rest are set to 0.3 in the two-phase multi-teacher RKD. the distillation frequency $N = min(10, 2 \times |\hat{R}^k|/10))$.

4.5. Main Results

We compare the performance of different methods using the same setting as ERDA (Qin and Joty, 2022). The reported scores are the average accuracy of 6 rounds.

FewRel Benchmark We report our results on 10-way 5-shot in table 1, while fig. 3 shows the results on the 10-way 2-shot and 10-way 10-shot settings. From the results, we can observe that:

(i) Our proposed RK2DA consistently outperforms existing baselines, achieving state-of-the-art performance across all CFRE settings. These results highlight the effectiveness of combining knowledge transfer from previous tasks and prototype augmentation. The approach enables the model to maintain a stable understanding of history relations, leading to improved performance.

(ii) Suprisingly, RK2DA outperforms all other methods in the first task of all settings. This superior performance can be attributed to the effective utilization of prototype data augmentation. By incorporating additional augmented data during training, RK2DA mitigates the risk of overfitting and achieves improved results in current task learning.

(iii) With the increase in data volume, the performance of various methods except SeqRun has improved. In both the 10way-2shot and 10way-5shot settings, RK2DA outperforms Joint-Training by a significant margin. However, in the 10way-10shot setting, RK2DA performs slightly worse. This discrepancy may arise from the difference between the distribution of pseudo-data and real data in the embedding space, which introduces bias during learning. Nonetheless, RK2DA still partially surpasses and approaches Joint Training which is obviously better than ERDA. This highlights the advantages of RK2DA in few-shot scenarios, as it can take full advantage of knowledge from previous tasks and minimal memorized data to retain relatively stable performance in continual few-shot learning.

(iv) Compared to ERDA, our prototype augmentation method is simpler yet it yields superior results across all experimental settings. This demonstrates that prototype augmentation effectively leverages limited data information for learning. In 10-way 5-shot settings, RK2DA outperforms ERDA by an average of 3.1% of accuracy. Furthermore, even as the task progresses, RK2DA maintains a distinct advantage over ERDA, showcasing its ability to maintain the stability of embedding space and facilitate learning of subsequent tasks. **TACRED Benchmark** The results in fig. 4 demonstrate the performance of the models trained with 5-way 5-shot and 5-way 10-shot on the TACRED dataset. Though TACRED is considered as a challenging task (Hu et al., 2022), we can observe that RK2DA achieves higher accuracy scores compared to all other methods and it approaches a performance level similar to Joint Training, which confirms the strong generalization ability of RK2DA.

Method	Task index							
	1	2	3	4	5	6	7	8
RK2DA-BT	**93.78**	81.88	72.88	66.96	63.02	59.74	57.19	53.99
RK2DA-Re	90.72	81.03	72.97	67.70	63.52	60.03	57.73	**54.73**
RK2DA-ProtoAug	93.57	82.38	73.49	68.25	63.94	59.76	56.40	53.69
RK2DA-KD	91.8	80.23	72.79	67.45	63.55	59.09	57.37	54.54
RK2DA-$\mathcal{L}_{RKD_1}$	93.12	82.40	74.35	68.63	64.39	60.20	57.37	54.04
RK2DA-$\mathcal{L}_{RKD_2}$	93.6	81.99	74.02	68.90	64.5	**60.72**	**57.82**	54.69
RK2DA	**93.78**	**83.05**	**74.67**	**69.52**	**64.83**	60.71	57.56	54.58

Table 2: Ablations on **FewRel** benchmark (10-way 5-shot).

4.6. Ablation Study

To validate the effectiveness and rationality of key components and steps of RK2DA, we conducted a series of ablation experiments on FewRel 10-way 5-shot setting. RK2DA's ablated variants include:(i) **RK2DA-BT** removes the Balance Tuning module. (ii) **RK2DA-Re** removes the Reconsolidation moudle.(iii) **RK2DA-ProtoAug** removes the prototype augmentation method, it dosen't add Gaussian noise from all modules and trained solely using real data. (iv) **RK2DA-KD** removes two-phase multi-teacher RKD method, Due to the absent of the knowledge distillation, it aslo removes the reconsolidation module. (v) **RK2DA-$\mathcal{L}_{RKD_1}$** removes the first phase of two-phase multi-teacher module RKD method. (vi) **RK2DA-$\mathcal{L}_{RKD_2}$** removes the second phase of two-phase multi-teacher module RKD method. From the results in table 2, we have the following analyses:

(i) Although **RK2DA-BT** performed well in the first task, it showed poor performance in the sub-

sequent tasks, proving that BT can alleviate the catastrophic of history relations. The performance of **RK2DA-Re** is poorer in previous tasks, but it showed better performance in $\mathcal{T}_7$ and $\mathcal{T}_8$. Our analysis suggests that the reconsolidation module aims to acquire knowledge of the current task, potentially sacrificing the accuracy of old tasks to achieve better performance in new task.

(ii) **RK2DA-ProtoAug** showed a decline in performance across all tasks. It indicates that utilizing pseudo data generated by Gaussian noise not only improves the learning of current relations but also mitigates overfitting on minority samples. **RK2DA-KD** underperforms on most tasks, suggesting the necessity of transferring knowledge from previous tasks to subsequent task learning processes. Meanwhile, the knowledge of the current task can facilitate the model acquire new knowledge in a stable embedding space. Through those two results, it becomes evident that RK2DA effectively integrates both techniques, thereby mutually enhancing their effectiveness.

(iii) **RK2DA-**$\mathcal{L}_{RKD_1}$ has lower accuracy than RK2DA, which demonstrates the importance of the first phase in our two-phase learning approach. The first phase helps transfer knowledge from previous tasks to maintain the stability of the embedding space, while **RK2DA-**$\mathcal{L}_{RKD_2}$ shows that the second phase assists in transferring knowledge from new tasks, thereby enhancing the learning of the current task. However, similar to **RK2DA-Re**, it also faces the trade-off of balancing performance improvements on the current task or maintaining accuracy on old tasks. This reflects the trade-off between stability and plasticity in CFRE and CRE (Zhang et al., 2022).

5. Conclusion

In this paper, we propose a novel method called RK2DA, to alleviate the catastrophic forgetting and overfitting problems which are the core issues in CFRE. RK2DA generates pseudo data points by introducing Gaussian noise to prototype embeddings to take full advantage of limited information. Moreover, RK2DA utilizes a novel two-phase multi-teacher relational knowledge distillation method to transfer various knowledge from different embedding spaces. Extensive experiments on two benchmark datasets demonstrate that our method significantly improved the performance compared to the most advanced methods. In future research, we aim to explore techniques for generating higher quality pseudo data to enhance the robustness of CFRE and investigate adaptive knowledge acquisition methods from diverse embedding spaces to further use the history knowledge.

Acknowledgments

We sincerely thank all anonymous reviewers for their valuable comments. This work is supported by the National Natural Science Foundation of China (No.62276095) and the National Social Science Foundation of China (No. 20&ZD047)

6. Bibliographical References

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905.

Richard Boyce, Stephen D Glasgow, Sylvain Williams, and Antoine Adamantidis. 2016. Causal evidence for the role of rem sleep theta rhythm in contextual memory consolidation. *Science (New York, N.Y.)*, page 812—816.

Jinxiu Chen, Donghong Ji, Chew Lim Tan, and Zhengyu Niu. 2006. Semi-supervised relation extraction with label propagation. In *Proceedings of the Human Language Technology Conference of the NAACL*, page 25–28.

Tianqi Chen, Ian Goodfellow, and Jonathon Shlens. 2016. Net2net: Accelerating learning via knowledge transfer.

Yung-Sung Chuang, Shang-Yu Su, and Yun-Nung Chen. 2020. Lifelong language knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2914–2924.

Li Cui, Deqing Yang, Jiaxin Yu, Chengwei Hu, Jiayang Cheng, Jingjie Yi, and Yanghua Xiao. 2021. Refining sample embeddings with relation prototypes to enhance continual 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*, pages 232–243.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. 2022. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 3366–3385.

Songlin Dong, Xiaopeng Hong, Xiaoyu Tao, Xinyuan Chang, Xing Wei, and Yihong Gong.

2021. Few-shot class-incremental learning via relation knowledge distillation. In *AAAI Conference on Artificial Intelligence*, pages 1255–1263.

Chrisantha Fernando, Dylan S. Banarse, Charles Blundell, Yori Zwols, David R Ha, Andrei A. Rusu, Alexander Pritzel, and Daan Wierstra. 2017. Pathnet: Evolution channels gradient descent in super neural networks. *ArXiv*.

Robert M. French. 1999. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, pages 128–135.

Tianyu Gao, Xu Han, Zhiyuan Liu, and Maosong Sun. 2019. Hybrid attention-based prototypical networks for noisy few-shot relation classification. pages 6407–6414.

Jianping Gou, B. Yu, Stephen J. Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, pages 1789 – 1819.

Xu Han, Yi Dai, Tianyu Gao, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2020. Continual relation learning via episodic memory activation and reconsolidation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6429–6440.

Xu Han, Pengfei Yu, Zhiyuan Liu, Maosong Sun, and Peng Li. 2018. Hierarchical relation extraction with coarse-to-fine grained attention. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2236–2245, Brussels, Belgium. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, pages 1735–1780.

Chengwei Hu, Deqing Yang, Haoliang Jin, Zhen Chen, and Yanghua Xiao. 2022. Improving continual relation extraction through prototypical contrastive learning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1885–1895.

Xuming Hu, Chenwei Zhang, Fukun Ma, Chenyao Liu, Lijie Wen, and Philip S. Yu. 2021. Semi-supervised relation extraction via incremental meta self-training. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 487–496.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, pages 3521–3526.

Zhizhong Li and Derek Hoiem. 2018. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 2935–2947.

ChunYang Liu, WenBo Sun, WenHan Chao, and WanXiang Che. 2013. Convolution neural network for relation extraction. In *Advanced Data Mining and Applications*, pages 231–242.

Arun Mallya, Dillon Davis, and Svetlana Lazebnik. 2018. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV*, page 72–88.

Michael McCloskey and Neal J. Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. Psychology of Learning and Motivation, pages 109–165.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. 2019. Relational knowledge distillation. In *CVPR*, pages 3962–3971.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Chengwei Qin and Shafiq Joty. 2022. Continual few-shot relation learning via embedding space regularization and data augmentation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 2776–2789.

Meng Qu, Tianyu Gao, Louis-Pascal Xhonneux, and Jian Tang. 2020. Few-shot relation extraction via Bayesian meta-learning on relation graphs. In *Proceedings of the 37th International Conference on Machine Learning*, pages 7867–7876.

Haopeng Ren, Yi Cai, Xiaofeng Chen, Guohua Wang, and Qing Li. 2020. A two-phase prototypical network model for incremental few-shot relation classification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1618–1629.

Haopeng Ren, Yi Cai, Raymond Y.K. Lau, Hofung Leung, and Qing Li. 2023. Granularity-aware area prototypical network with bimargin loss for few shot relation classification. *IEEE*

Transactions on Knowledge and Data Engineering, pages 4852–4866.

Hippolyt Ritter, Aleksandar Botev, and David Barber. 2018. Online structured laplace approximations for overcoming catastrophic forgetting. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, page 3742–3752.

Ang Sun, Ralph Grishman, and Satoshi Sekine. 2011. Semi-supervised relation extraction with large-scale word clustering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 521–529.

Quynh-Trang Pham Thi, Anh-Cuong Pham, Ngoc-Huyen Ngo, and Duc-Trong Le. 2022. Memory-based method using prototype augmentation for continual relation extraction. *2022 RIVF International Conference on Computing and Communication Technologies (RIVF)*, pages 1–6.

Giulio Tononi and Chiara Cirelli. 2006. Sleep function and synaptic homeostasis. *Sleep Medicine Reviews*, pages 49–62.

Hong Wang, Wenhan Xiong, Mo Yu, Xiaoxiao Guo, Shiyu Chang, and William Yang Wang. 2019. Sentence embedding alignment for lifelong relation extraction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 796–806.

Peiyi Wang, Yifan Song, Tianyu Liu, Rundong Gao, Binghuai Lin, Yunbo Cao, and Zhifang Sui. 2022. Less is more: Rethinking state-of-the-art continual relation extraction models with a frustratingly easy but effective approach.

Yushi Wang, Jonathan Berant, and Percy Liang. 2015. Building a semantic parser overnight. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1332–1342, Beijing, China. Association for Computational Linguistics.

Guang Yang, Cora Sau Wan Lai, Joseph Cichon, Lei Ma, Wei Li, and Wen Biao Gan. 2014. Sleep promotes branch-specific formation of dendritic spines after learning. *Science*, pages 1173–1178.

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

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.

Mo Yu, Wenpeng Yin, Kazi Saidul Hasan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2017. Improved neural relation detection for knowledge base question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 571–581.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2002. Kernel methods for relation extraction. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 71–78.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *International Conference on Computational Linguistics*.

Han Zhang, Bin Liang, Min Yang, Hui Wang, and Ruifeng Xu. 2022. Prompt-based prototypical framework for continual relation extraction. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pages 2801–2813.

Kang Zhao, Hua Xu, Jiangong Yang, and Kai Gao. 2022. Consistent representation learning for continual relation extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3402–3411.

Fei Zhu, Xu-Yao Zhang, Chuan Wang, Fei Yin, and Cheng-Lin Liu. 2021. Prototype augmentation and self-supervision for incremental learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5867–5876.

7. Language Resource References

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.

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.