Learning Robust Representations for Continual Relation Extraction via Adversarial Class Augmentation

Peiyi Wang[1*] **Yifan Song**[1*] **Tianyu Liu**[2] **Binghuai Lin**[2]
Yunbo Cao[2] **Sujian Li**[1] **Zhifang Sui**[1]
[1] MOE Key Laboratory of Computational Linguistics, Peking University, China
[2] Tencent Cloud Xiaowei
wangpeiyi9979@gmail.com; {yfsong, lisujian, szf}@pku.edu.cn
{rogertyliu, binghuailin, yunbocao}@tencent.com;

Abstract

Continual relation extraction (CRE) aims to continually learn new relations from a class-incremental data stream. CRE model usually suffers from catastrophic forgetting problem, i.e., the performance of old relations seriously degrades when the model learns new relations. Most previous work attributes catastrophic forgetting to the corruption of the learned representations as new relations come, with an implicit assumption that the CRE models have adequately learned the old relations. In this paper, through empirical studies we argue that this assumption may not hold, and an important reason for catastrophic forgetting is that the learned representations do not have good robustness against the appearance of analogous relations in the subsequent learning process. To address this issue, we encourage the model to learn more precise and robust representations through a simple yet effective adversarial class augmentation mechanism (ACA), which is easy to implement and model-agnostic. Experimental results show that ACA can consistently improve the performance of state-of-the-art CRE models on two popular benchmarks. Our code is available at https://github.com/Wangpeiyi9979/ACA.

1 Introduction

Relation extraction (RE) aims to detect the relation of two given entities in a sentence. Traditional RE models are trained on a fixed dataset with a predefined relation set, which cannot handle the real-life situation where new relations are constantly emerging. To this end, continual relation extraction (CRE) (Wang et al., 2019; Han et al., 2020; Cui et al., 2021; Zhao et al., 2022; Wang et al., 2022) is introduced. As shown in Figure 1, CRE is formulated as a class-incremental problem, which trains the model on a sequence of tasks. In each task, the model needs to learn some new relations and is evaluated on all seen relations. Like other continual learning systems, CRE models also suffer from catastrophic forgetting, i.e., the performance of previously learned relations seriously degrades when learning new relations.

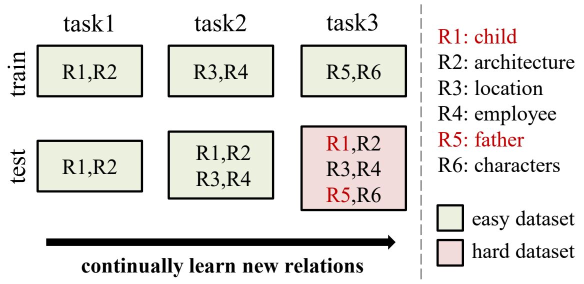


Figure 1: A demonstration for continual relation extraction with three tasks where each task involves two new relations. The learned representations from the easy training task can not handle the hard testing data, which contains analogous relations inherently hard to distinguish, e.g., "child" and "father".

The mainstream research in CRE (Han et al., 2020; Cui et al., 2021; Zhao et al., 2022; Wang et al., 2022) mainly attributes catastrophic forgetting to the corruption of the learned knowledge as new tasks come. To this end, a variety of sophisticated rehearsal-based mechanisms are introduced to better retain or recover the knowledge, such as relation prototypes (Han et al., 2020; Cui et al., 2021), curriculum-meta learning (Wu et al., 2021), contrastive replay and knowledge distillation (Zhao et al., 2022). All these methods implicitly assume that the model has adequately learned old relations. However, in this paper, we find that this assumption may not hold.

With a series of empirical studies, we observe that catastrophic forgetting mostly happens on some specific relations, and significant performance degradation tends to occur when their analogous relations appear. Based on our observations, we find another reason for catastrophic forgetting,

[*]Equal contribution.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 6264–6278
December 7-11, 2022 ©2022 Association for Computational Linguistics

i.e., *CRE models do not learn sufficiently robust representations of relations in the first place due to the relatively easy training task*. Taking "child" in Figure 1 as an example, because of the absence of hard negative classes in task 1, the CRE model tends to rely on shortcuts, such as entity types, to identify "child". Although the learned imprecise representations can handle the testing set of task 1 and task 2, they are not robust enough to distinguish "child" from its analogous relation ("father") in task 3. Therefore, the performance of "child" will decrease significantly when "father" appears. In contrast, relations such as "architecture" still perform well in task 3 because their analogous relations have not yet appeared.

Recently, adversarial data augmentation emerges as a strong baseline to prevent models from learning shortcuts from the easy dataset (Volpi et al., 2018; Zhao et al., 2020; Hendrycks et al., 2020). Inspired by these work, we introduce a simple yet effective Adversarial Class Augmentation (ACA) mechanism to improve the robustness of the CRE model. Concretely, ACA utilizes two class augmentation methods, namely hybrid-class augmentation and reversed-class augmentation, to build hard negative classes for new tasks. When a task arrives, ACA jointly trains new relations with adversarial augmented classes to learn robust representations. Note that our method is orthogonal to all previous work: ACA focuses on learning knowledge of newly emerging relations better, while previous methods are proposed to retain or recover learned knowledge of old relations[1]. Therefore, incorporating ACA into previous CRE models can further improve their performance.

We summarize our contributions as: **1)** we conduct a series of empirical studies on two strong CRE methods and observe that catastrophic forgetting is strongly related with the existence of analogous relations. **2)** we find an important reason for catastrophic forgetting in CRE which is overlooked in all previous work: the CRE models suffer from learning shortcuts to identify new relations, which are not robust enough against the appearance of their analogous relations. **3)** we propose an adversarial class augmentation mechanism to help CRE models learn more robust representations. Experimental results on two benchmarks show that our method can consistently improve the performance of two state-of-the-art methods.

2 Related Work

Relation Extraction Conventional Relation Extraction (RE) focuses on extracting the predefined relation of two given entities in a sentence. Recently, a variety of deep neural networks (DNN) have been proposed for RE, mainly including: **1)** Convolutional or Recurrent neural network (CNN or RNN) based methods (dos Santos et al., 2015; Wang et al., 2016; Xiao and Liu, 2016; Liu et al., 2019), which can effectively extract textual features. **2)** Graph neural network (GNN) based methods (Xu et al., 2015, 2016; Cai et al., 2016; Mandya et al., 2020), which jointly encode the sentence with lexical features. **3)** Pre-trained language model (PLM) based methods (Baldini Soares et al., 2019; Peng et al., 2020), which achieve state-of-the-arts on RE task.

Continual Learning Continual Learning (CL) aims to continually accumulate knowledge from a sequence of tasks (De Lange et al., 2019). A major challenge of CL is catastrophic forgetting, i.e., the performance of previously learned tasks seriously drops when learning new tasks. To this end, prior CL methods can be roughly divided into three groups: **1)** Rehearsal-based methods (Rebuffi et al., 2017; Wu et al., 2019) maintain a memory to save instances of previous tasks and replay them during training of new tasks. **2)** Regularization-based methods (Kirkpatrick et al., 2017; Aljundi et al., 2018) add constraints on the weights important to old tasks. **3)** Architecture-based methods (Mallya and Lazebnik, 2018; Qin et al., 2021) dynamically change model architectures to learn new tasks and prevent forgetting old tasks. Recently, rehearsal-based methods have been proven to be the most effective for NLP tasks, including relation extraction. We focus on the rehearsal-based methods for CRE in this paper.

Shortcuts Learning Phenomenon Shortcuts learning phenomenon denotes that DNN models tend to learn unreliable shortcuts in datasets, leading to poor generalization ability in real-world applications (Lai et al., 2021). Recently, researchers have revealed the shortcut learning phenomenon for different kinds of language tasks, such as natural language inference (He et al., 2019), information

[1] The proposed method can be viewed as a "precaution" that takes place in the current task to mitigate the catastrophic forgetting on the analogous relations in the subsequent tasks. While the prior work is more like a "remedy" for the current task to recall the already learned knowledge in the past tasks.

extraction (Wang et al., 2021), reading comprehension (Lai et al., 2021) and question answering (Mudrakarta et al., 2018). Geirhos et al. (2020) points out that shortcuts learning phenomenon happens because the "Principle of Least Effort" (Kingsley, 1972), i.e., people (also animal and machine) will naturally minimize the amount of effort to solve tasks. Recently, data augmentation (Tu et al., 2020) and adversarial training (Stacey et al., 2020) are used to alleviate shortcuts learning phenomenon with synthesized data. To the best of our knowledge, we are the first work to analyze the catastrophic forgetting in CRE from the perspective of shortcuts learning, and propose an adversarial data augmentation method to alleviate it.

3 Task Formulation

In CRE, the model is trained on a sequence of tasks $(T_1, T_2, ..., T_k)$. Each task T_i can be represented as a triplet (R_i, D_i, Q_i), where R_i is the set of new relations, D_i and Q_i are the training and testing set, respectively. Every instance $(x_j, y_j) \in D_i \cup Q_i$ belongs to a specific relation $y_j \in R_i$. The goal of CRE is to continually train the model on new tasks to learn new relations, while avoiding forgetting of previously learned ones. More formally, in the i-th task, the model learns new relations R_i from D_i, and should be able to identify all seen relations, i.e., the model will be evaluated on the all seen testing sets $\bigcup_{j=1}^{i} Q_j$. To alleviate catastrophic forgetting in CRE, previous work (Cui et al., 2021; Han et al., 2020; Zhao et al., 2022; Wang et al., 2022) adopts a memory to store a few typical instances (e.g., 10) for each old relation. In the subsequent training process, instances in the memory will be replayed to alleviate the catastrophic forgetting.

4 Catastrophic Forgetting in CRE: Characteristics and the Cause

In this section, we conduct a series of empirical studies on two state-of-the-art CRE models, namely EMAR (Han et al., 2020) and RP-CRE (Cui et al., 2021), and two benchmarks, namely FewRel and TACRED, to analyze catastrophic forgetting in CRE. Please refer to Section 6.1 for details of these two benchmarks and two CRE models.

4.1 Characteristics of Catastrophic Forgetting

We use *Forgetting Rate* (FR) (Chaudhry et al., 2018a,b) to measure the average forgetting of a relation. Assuming that relation r appears in task i, the FR of r after the model has finished the tasks sequence $(T_1, ..., T_i, ..., T_k)$ is defined as:

$$FR_r = \frac{1}{k-i} \sum_{j=i+1}^{k} pd_r^j \tag{1}$$

$$pd_r^j = \max_{l \in \{i, ..., j-1\}} F1_r^l - F1_r^j, \tag{2}$$

where pd_r^j and $F1_r^j$ are the performance degradation and F1 score of r after the model trains on task j, respectively. The sequence length k is 10 for both FewRel and TACRED.

Model	ID	FR (%)	MS	F1	F1* (Δ)
EMAR	G1	1.3	0.42	95.4	97.4 (↑ 2.0)
	G2	4.5	0.53	84.6	90.9 (↑ 6.3)
	G3	9.4	0.62	69.8	81.5 (↑ 11.7)
RP-CRE	G1	1.2	0.42	95.5	97.4 (↑ 1.9)
	G2	4.7	0.53	83.8	90.8 (↑ 7.0)
	G3	9.9	0.63	69.5	81.5 (↑ 12.0)

Table 1: We divide relations of FewRel into three groups according to their forgetting rate (FR). "MS" is short for *max similarity*. F1 and F1* are the Macro-F1 scores of EMAR/RP-CRE and the supervised model which trains all data together, respectively. Δ is the performance gap between two CRE models and the supervised model.

We divide all relations into three equal-sized groups based on their FR from small to large. As shown in Table 1, relations in G1 hardly suffer from forgetting as FR is just 1.3% and 1.2% on EMAR and RP-CRE, respectively. In contrast, relations in G3 have catastrophic forgetting and the FR is close to 10% for both models. A similar tendency is observed on TACRED (please refer to Appendix A for details). To explore why FR varies widely among different relations, we dive into the results of two CRE models and ask two questions.

Where catastrophic forgetting happens? With careful comparison between G1 and G3, we find that relations in G3 seem to have analogous relations in the dataset. For example, "mother" belongs to G3, and there are its semantically analogous relations, such as "spouse", in the dataset. To confirm our finding, we first define the similarity for a pair of relations as the cosine distance of their prototypes, i.e., the mean vanilla BERT sentence embedding of all corresponding instances. Then, for a certain relation, we compute its *max similarity* (MS) to all the other relations in the dataset. As shown in Table 1, MS of G3 is significantly greater than that of G1, indicating that the catastrophic

Models	FewRel		TACRED	
	#CF	#SIM	#CF	#SIM
EMAR	666	614 (92%)	766	690 (90%)
RP-CRE	781	688 (88%)	949	773 (81%)

Table 2: Analysis of catastrophic forgetting on 50 different runs. "#CF" denotes the total number of catastrophic forgetting cases, and "#SIM" are bad cases accompanied by the appearance of top-5 analogous relations.

Models	FewRel			TACRED		
	G1	G2	G3	G1	G2	G3
EMAR	95.6	73.9	49.3	69.8	46.1	30.1
RP-CRE	94.7	75.7	51.6	75.3	48.5	36.6
Sup.	99.7	95.6	86.8	94.0	84.2	68.9

Table 3: Retrieval precision of different methods on two benchmarks. We divide all relations into three groups according to their forgetting rate the same as Table 1.

forgetting mostly happens on relations with large MS[2]. Besides, as shown in the last two columns (F1 and F1$^*(\Delta)$) of Table 1, we also observe that the performance gap between CRE models and the supervised model significantly grows as MS increases, showing that CRE poses a more serious challenge to identify the relations with large MS.

When catastrophic forgetting happens? We also observe that the performance of the relations with high FR always has a sudden drop in some tasks. To explore the characteristic of the task with severe performance drop, we run two CRE models on 50 different task sequences, and record all the bad cases where catastrophic forgetting happens (the F1 scores of a relation degrades greater than 10 points after the model learns a new task). Given a certain relation r and its corresponding bad cases, we mark cases where exist top-5 most similar relations of r. As shown in Table 2, we observe that more than 80% bad cases on both benchmarks are related to the appearance of top-5 most similar relations. Taking relation "mother" in FewRel dataset as an example, more than 90% bad cases contain relation "spouse", which has the top-1 similarity with "mother"[3]. These results show that for a relation that suffers catastrophic forgetting, significant performance degradation is usually accompanied by the appearance of their analogous relations..

4.2 The Cause of Catastrophic Forgetting

All of the previous CRE works attribute the catastrophic forgetting to the corruption of the learned knowledge during the continual learning process, with the assumption that the CRE models have adequately learned the previous relations. However, we argue that this assumption may not hold. In CRE, models are continually trained on a sequence of stand-alone easy training datasets, where each dataset usually only consists of very few new relations without analogous relations appearing together. In contrast, CRE models are evaluated on the much harder testing dataset of all seen relations, which usually contains analogous relations. He et al. (2019); Karimi Mahabadi et al. (2020); Minderer et al. (2020) find that the deep neural network tends to learn shortcuts in the simple training dataset to make the decision, leading to poor generalization. Therefore, we point out another important reason for the performance degradation of learned relations: *the CRE models suffer from learning shortcuts in the easy training dataset to identify relations, which are not robust enough against the appearance of their analogous relations.* This reason can well explain our observed phenomena, i.e, catastrophic forgetting mostly happens on some specific relations with analogous relations in the subsequent tasks, and significant performance degradation nearly always happens when their analogous relations appear.

To confirm our hypothesis, we propose a retrieval test: after the CRE model is trained to identify a specific relation r, we use the trained model to retrieve instances of r from the whole test set according to the similarity of representations[4]. If the learned representations are not robust enough, the corresponding retrieval precision will be relatively low. Table 3 shows the result of the retrieval results of two CRE models and the supervised model. Compared with the supervised model, two CRE methods retrieve much more unrelated instances, especially for relations suffering severe forgetting, showing that the CRE models indeed learn representations that lack robustness.

5 Methodology

Recently, adversarial data augmentation has shown promise for avoiding models from learning shortcuts in the easy dataset (Volpi et al., 2018; Zhao et al., 2020; Hendrycks et al., 2020; Zhu et al.,

[2] Appendix G provides more details of relations in G1/G3.

[3] Details of some bad cases can be founded in Appendix B.

[4] Please refer to Appendix C for details of our retrieval test.

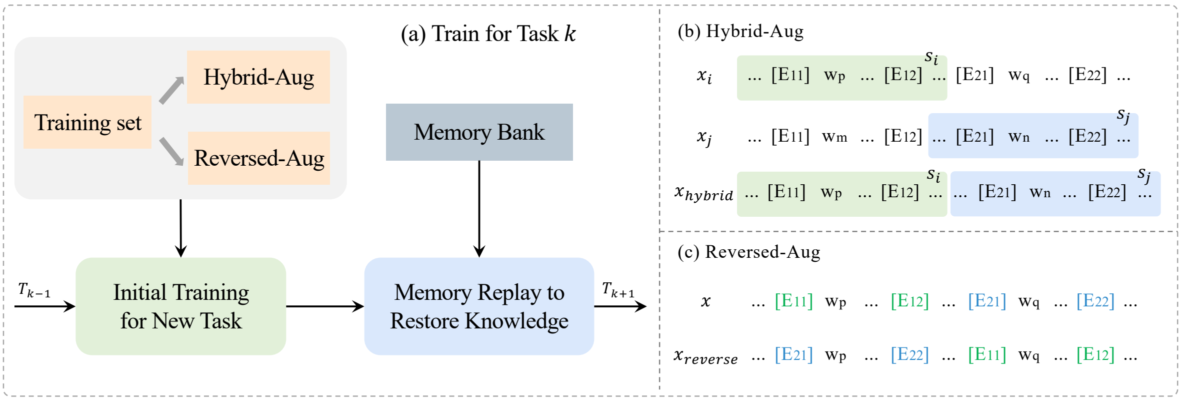

(a) Train for Task k
Training set
Hybrid-Aug
Reversed-Aug
Memory Bank
T_{k-1}
Initial Training for New Task
Memory Replay to Restore Knowledge
T_{k+1}
(b) Hybrid-Aug
x_i ... [E11] wp ... [E12] ... [E21] wq ... [E22] ...
s_i
x_j ... [E11] wm ... [E12] ... [E21] wn ... [E22] ...
s_j
x_{hybrid} ... [E11] wp ... [E12] ... ... [E21] wn ... [E22] ...
(c) Reversed-Aug
x ... [E11] wp ... [E12] ... [E21] wq ... [E22] ...
$x_{reverse}$ ... [E21] wp ... [E22] ... [E11] wq ... [E12] ...

Figure 2: (a) A demonstration for learning process of existing typical CRE models with our adversarial class augmentation mechanism. (b) Hybrid-class augmentation. (c) Reversed-class augmentation. We use "[E_{11}]/[E_{12}]" and "[E_{21}]/[E_{22}]" to mark the head entity e^1 and tail entity e^2, respectively.

2021). Therefore, in this section, we propose a simple yet effective adversarial class augmentation mechanism (ACA) containing two kinds of class augmentation to help the CRE model learn more robust representations.

5.1 Two-Stage Training

Our ACA is model-agnostic and utilizes popular state-of-the-art CRE models as the backbone. Therefore, we first briefly introduce the two-stage training process of these CRE models.

CRE model aim to finish a sequence of tasks $(T_1, T_2, ..., T_k)$. Without loss of generality, we represent CRE model with two components: **1)** an encoder, which maps an input instance x into a representation vector; **2)** a classifier, which produces a probability distribution over all seen relations till current task as the prediction for x. As shown in Figure 2(a), previous CRE methods (Han et al., 2020; Cui et al., 2021; Zhao et al., 2022; Wang et al., 2022) can be generally formulated as a two-stage training process. **1)** initial training: they first expand the class node in the classifier for new relations, and then train the CRE model with only new data to learn new relations; **2)** memory replay: they first update the memory bank with new data, and then replay the memory bank to restore the knowledge of previously learned relations. Specifically, previous work mainly focuses on the memory replay stage and proposes various sophisticated mechanisms to better retain or recover the learned knowledge, while improvements to the initial training stage remain under-explored. See more details of these CRE methods in their original paper.

5.2 Adversarial Class Augmentation

Orthogonal to all previous CRE models, our ACA instead focuses on the first initial training stage to improve the robustness of newly learned relation representations. Specifically, when a new task T_i comes, ACA first augments the new relations R_i based on the new training set D_i, and then trains the original relations and synthesized classes together.

Hybrid-class Augmentation Given N new relations, we pair them randomly and get $\lfloor N/2 \rfloor$ relation pairs. We construct hybrid synthetic classes based on these relation pairs. Specifically, for a relation pair $\{r_i, r_j\}$ with the relations r_i and r_j, we use two instances, x_i from r_i and x_j from r_j, to generate a hybrid instance x_{hybrid} for the extra synthetic class r_{ij}. As shown in Figure 2(b), we first extract a span s_i that contains the head entity e_i^1 but excludes the tail entity e_i^2 from x_i, and a span s_j that contains the tail entity e_j^2 but excludes the head entity e_j^1 from x_j, and then concatenate s_i and s_j to form $x_{hybrid} = [s_i; s_j]$. Through the hybrid-class augmentation, we can construct $\lfloor N/2 \rfloor$ extra classes for the current new task.

Reversed-class Augmentation We classify all relations into two categories, symmetric and asymmetric relations. The symmetric relation means that the order of the head and tail entities does not matter, e.g, "sibling" and "spouse" (please refer to Appendix E for details of symmetric relations on two datasets). In contrast, the semantic of the asymmetric relations is related to the choice of head and tail entities, e.g., "located in" and "mother". As shown in Figure 2(c), reversed-class augmentation reverses the head and tail entities of each asymmet-

FewRel

Models	**T1**	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**	**T10** (Δ)
EA-EMR (Wang et al., 2019)	89.0	69.0	59.1	54.2	47.8	46.1	43.1	40.7	38.6	35.2 (–)
CML (Wu et al., 2021)	91.2	74.8	68.2	58.2	53.7	50.4	47.8	44.4	43.1	39.7 (–)
RPCRE (Cui et al., 2021)	97.9	92.7	91.6	89.2	88.4	86.8	85.1	84.1	82.2	81.5 (–)
CRL (Zhao et al., 2022)	98.2	94.6	92.5	90.5	89.4	87.9	86.9	85.6	84.5	83.1 (–)
RP-CRE† (Cui et al., 2021)	97.8	94.7	92.1	90.3	89.4	88.0	87.1	85.8	84.4	82.8 (–)
RP-CRE† + **ACA**	98.0	94.7	92.2	90.6	89.9	89.0	87.5	86.4	85.7	83.8 (↑ 1.0)
EMAR† (Han et al., 2020)	98.1	94.3	92.3	90.5	89.7	88.5	87.2	86.1	84.8	83.6 (–)
EMAR† + **ACA**	**98.3**	**95.0**	**92.6**	**91.3**	**90.4**	**89.2**	**87.6**	**87.0**	**86.3**	**84.7** (↑ 1.1)

TACRED

Models	**T1**	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**	**T10** (Δ)
EA-EMR (Wang et al., 2019)	47.5	40.1	38.3	29.9	24.0	27.3	26.9	25.8	22.9	19.8 (–)
CML (Wu et al., 2021)	57.2	51.4	41.3	39.3	35.9	28.9	27.3	26.9	24.8	23.4 (–)
RP-CRE (Cui et al., 2021)	97.6	90.6	86.1	82.4	79.8	77.2	75.1	73.7	72.4	72.4 (–)
CRL (Zhao et al., 2022)	97.7	93.2	89.8	84.7	84.1	81.3	**80.2**	**79.1**	**79.0**	78.0 (–)
RP-CRE† (Cui et al., 2021)	97.5	92.2	89.1	84.2	81.7	81.0	78.1	76.1	75.0	75.3 (–)
RP-CRE† + **ACA**	97.8	**93.6**	89.9	84.4	82.7	81.1	78.2	77.7	75.5	76.2 (↑ 0.9)
EMAR† (Han et al., 2020)	**98.3**	92.0	87.4	84.1	82.1	80.6	78.3	76.6	76.8	76.1 (–)
EMAR† + **ACA**	98.0	92.1	**90.6**	**85.5**	**84.4**	**82.2**	80.0	78.6	78.8	**78.1** (↑ 2.0)

Table 4: Accuracy (%) on all seen relations at the stage of learning current tasks. We report the average accuracy of 5 different runs. † denotes our reproduced results with the open codebases. Other results are directly taking from Zhao et al. (2022). Δ is the performance gap in T10 between original CRE models and the models with our adversarial class augmentation mechanism (+ **ACA**). EMAR and RP-CRE with ACA significantly outperform their corresponding vanilla models ($p < 0.05$).

rical relation to construct the extra classes.

Adversarial Training Given N new relations, we can generate at most $N + \lfloor N/2 \rfloor$ hard negative classes using the two augmentation methods. Thus, in the initial training stage, the model is jointly trained to classify $(2N + \lfloor N/2 \rfloor)$ classes to better learn the original new relations. At the end of initial training, the extended class nodes of augmented classes in the classifier will be removed.

6 Experiments

6.1 Experimental Setups

Datasets Following previous works (Han et al., 2020; Wu et al., 2021; Cui et al., 2021; Zhao et al., 2022), our experiments are conducted upon two widely used datasets, **FewRel** (Han et al., 2018) and **TACRED** (Zhang et al., 2017). Please refer to Appendix D for details of these two datasets. We construct 5 different task sequences for both FewRel and TACRED. For each task sequence, we simulate 10 tasks by randomly dividing all relations of the dataset into 10 sets. For a fair comparison, our 5 task sequences are exactly the same as that of Cui et al. (2021) and Zhao et al. (2022).

Evaluation Metrics Following Cui et al. (2021) and Zhao et al. (2022), we use average accuracy on all seen tasks as our evaluation metric. For a stronger method, the average accuracy of each task should be consistently higher than that of baselines.

Baselines We consider the following baselines: **EA-EMR** (Wang et al., 2019), which maintains a memory replay and embedding alignment mechanism to alleviate catastrophic forgetting; **CML** (Wu et al., 2021), which introduces curriculum learning and meta-learn to alleviate catastrophic forgetting in CRE; **EMAR** (Han et al., 2020), which proposes a memory activation and reconsolidation mechanism to retain the learned knowledge. Note that the original EMAR was based on a Bi-LSTM encoder, and we re-implement EMAR with BERT; **RP-CRE** (Cui et al., 2021), which proposes a memory network to retain the learned representations with relation prototypes; **CRL** (Zhao et al., 2022), which adopts contrastive learning replay and knowledge

Models	FewRel	TACRED
EMAR+ACA	84.7	78.1
w/o hybrid-class aug.	84.3	77.4
w/o reversed-class aug.	83.9	76.8
w/o both classes aug.	83.6	76.1

Table 5: The effect of two class augmentation methods on two benchmarks. "aug." is short for augmentation.

distillation to retain the learned knowledge.

Implement Details Our ACA is model-agnostic, and we choose two state-of-the-art CRE models, EMAR and RP-CRE as our backbone to evaluate ACA. The number of stored instances of each relation in the memory bank is 10. All hyperparameters of EMAR and RP-CRE are the same as that of their origin paper. ACA does not introduce any model hyperparameters. We run our code on a single NVIDIA A40 GPU with 48GB memory, and report the average result of 5 different task sequences.

6.2 Main Results

The performances of our ACA and baselines are shown in Table 4. As shown, after applying ACA, the performances of EMAR and RP-CRE consistently improve in nearly all training stages of two benchmarks. Previous CRE work usually regards the accuracy of the last task as the most important metric. For the accuracy of T10, our proposed ACA improves RP-CRE/EMAR by 1.0/1.1 and 0.9/2.0 accuracy on FewRel and TACRED, respectively. Furthermore, EMAR+ACA achieves new state-of-the-art results on both two benchmarks. These results demonstrate the effectiveness and universality of our proposed method.

7 Analysis

7.1 Ablation Study

To further explore the effectiveness of our proposed two class augmentation methods, we conduct an ablation study. Table 5 shows the results of EMAR with different augmentation methods on two benchmarks. We find that both augmentations are conducive to the model performance, and they are complementary to each other. In addition, the reversed-class augmentation is more effective than the hybrid-class augmentation. We think the reason is that the reversed-class augmentation can maintain the fluency of the constructed sentences, while the hybrid-class augmentation cannot.

Models	FewRel			TACRED		
	G1	G2	G3	G1	G2	G3
EMAR	95.6	73.9	49.3	69.8	46.1	30.1
EMAR+ACA	97.3	83.0	57.7	77.9	52.6	33.5

Table 6: Retrieval precision of different models on FewRel. The results are divided into three groups according to their forgetting rate of EMAR.

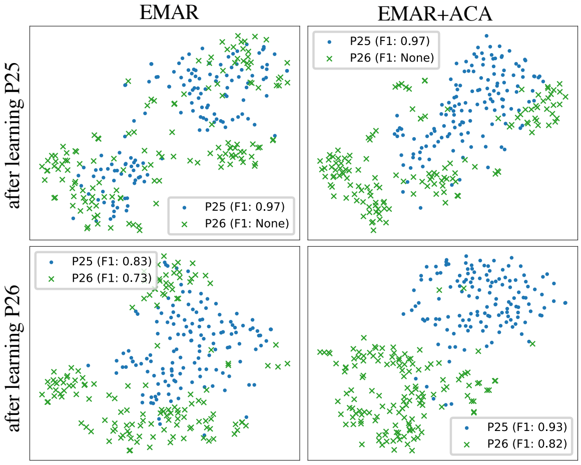


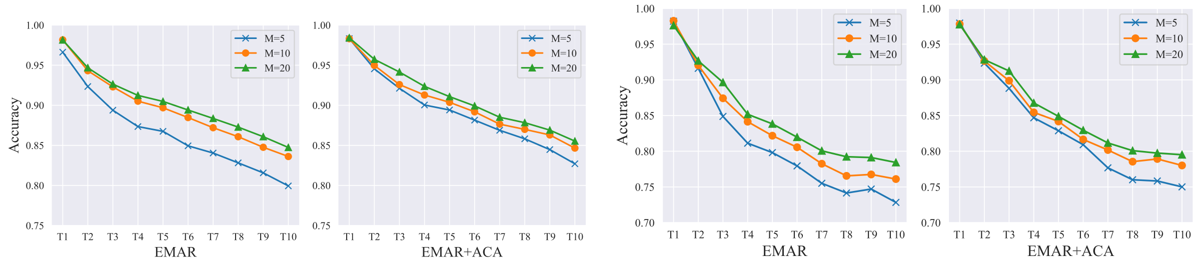
Figure 3: The representation of instances belonging to P25 ("mother") and P26 ("spouse") after learning P25 and P26, respectively.

7.2 Robust Representation Learning

Our proposed ACA aims to learn robust representations that can better distinguish analogous relations. To further confirm the effectiveness of our method, we first reproduce the retrieval test introduced in our pilot experiments (see Appendix C for more details). Table 6 shows the results of EMAR and EMAR+ACA on two benchmarks. As is shown, ACA can significantly increase the precision of retrieval results, showing that our method indeed helps the model learn more robust representations.

We also conduct a case study to intuitively show the effectiveness of our method. We consider two analogous relations, P25 ("mother") and P26 ("spouse")[5], and EMAR catastrophically forgets P25 when P26 appears. We use t-SNE to visualize the representation of all instances belonging to these two relations after the model learning P25 and P26, respectively. As shown in Figure 3: **1)** For EMAR, after the model learns the relation P25, it relies on shortcuts, such as entity types, to identify instances of P25. Therefore, the representations of the instances belonging to P25 and P26 are mixed together, which means the learned rep-

[5] Please refer to Appendix F for more cases.



(a) Results on FewRel

(b) Results on TACRED

Figure 4: Comparison of model's dependence on memory size. We report the average results of 5 different runs. ACA improves the stability of EMAR on two benchmarks.

resentation of P25 can also represent instances of P26. When P26 appears, it is hard to learn a more robust representation of P25 with P26 with only very limited memory instances of P25, and thus EMAR catastrophically forgets P25 (the F1 score of P25 significantly degrades 14 points); **2)** For EMAR+ACA, when learning P25, the model can learn more robust representations of P25 with our augmented relations. Thus, the representation of instances belonging to P25 and P26 is much more separable than that of EMAR. When P26 appears, the forgetting problem of P25 is greatly alleviated (the F1 score of P25 only drops 4 points).

7.3 Influence of Memory Size

Memory size is the number of memorized instances for each relation, which is a key factor for the model performance of rehearsal-based CRE methods. Therefore, in this section, we study the influence of memory size on our ACA.

We compare the performance of EMAR and EMAR+ACA with memory sizes 5, 10 and 20. As shown in Figure 4: **1)** As the size of the memory decreases, the performances of both models drop, showing the importance of the memory size for CRE models; **2)** On both FewRel and TACRED, EMAR+ACA outperforms EMAR under all three different memory sizes, further demonstrating the effectiveness of our ACA; **3)** As memory size decreases, EMAR+ACA shows a relatively stable performances. Specifically, EMAR+ACA outperforms EMAR 2.8, 1.1 and 0.7 accuracy on FewRel when the memory size is 5, 10, 20, respectively. A similar tendency is also observed on TACRED. These results further demonstrate the effectiveness of robust representations learned through our ACA.

GROUP			EMAR		+ACA	
ID	**MS**	**F1***	**FR**	**F1**	**FR**	**F1 (Δ)**
G1	0.39	95.0	2.5	91.2	2.6	91.4 (↑ 0.2)
G2	0.51	90.8	5.2	83.3	4.9	84.8 (↑ 0.5)
G3	0.67	84.0	7.5	75.3	7.1	76.2 (↑ 0.9)

Table 7: We equally divide relations of FewRel into 3 groups by their *max similarity* (MS). F1* is the F1 score of the supervised model. Δ denotes the performance gap between EMAR and EMAR+ACA.

7.4 Error Analysis

In this section, we conduct an error analysis to show the effectiveness of ACA and the challenge of CRE. Through our analysis of catastrophic forgetting, we find that the performance of relations is highly related to their *max similarity*. Therefore, we equally divide the relations into three groups according to their *max similarity*. As shown in Table 7: **1)** ACA mainly improves the performance and reduces the forgetting rate of relations with large *max similarity*; **2)** although ACA is efficient, relations with large *max similarity* still suffer from catastrophic forgetting and have a large performance gap with the supervised model. Therefore, future work should pay more attention to these relations.

8 Conclusion

In this paper, we conduct a series of empirical study to analyze catastrophic forgetting in CRE, and observe that catastrophic forgetting mostly happens on some specific relations, and significant performance degradation tends to occur when their analogous relations appear in subsequent tasks. Based on our observations, we find an important reason for catastrophic forgetting in CRE that all previous works overlooked, i.e., the CRE models suffer from learning shortcuts to identify new relations, which are not robust enough against the appearance of their analogous relations. To this end, we propose a simple yet effective adversarial class augmentation mechanism to help CRE models learn more robust representations. Extensive experiments on two benchmarks show that our method can further improve the performance of two state-of-the-art CRE models.

Limitations

Our paper has several limitations: **1)** Although we provide a new perspective from the shortcut learning to explain catastrophic forgetting, and utilize a retrieval test to confirm our hypothesis, we do not explore which types of shortcuts are learned by CRE models; **2)** Our ACA with two class augmentation methods is specially designed for CRE. However, our findings about catastrophic forgetting in this paper may be common in the context of continual learning. Therefore, it would be better if we can propose more universal adversarial training methods which can be adapted to all continual learning systems; **3)** ACA conducts the class augmentation before the initial training stage, which introduces extra computational overhead on top of backbone CRE models.

Acknowledgements

We thank Dawei Zhu and Zhejian Zhou for proofreading the paper and providing insightful comments. We thank Lei Li for some useful discussions. We also thank the anonymous reviewers for their thoughtful and constructive comments. This work is supported by National Key Research and Development Project (2019YFB1704002) and NSFC project U19A2065.

References

Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. 2018. Memory Aware Synapses: Learning What (not) to Forget. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, volume 11207, pages 144–161. Cham.

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, Florence, Italy.

Rui Cai, Xiaodong Zhang, and Houfeng Wang. 2016. Bidirectional Recurrent Convolutional Neural Network for Relation Classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 756–765, Berlin, Germany.

Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. 2018a. Riemannian Walk for Incremental Learning: Understanding Forgetting and Intransigence. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, volume 11215, pages 556–572. Cham.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. 2018b. Efficient Lifelong Learning with A-GEM. In *International Conference on Learning Representations*.

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 (Volume 1: Long Papers)*, pages 232–243, Online.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. 2019. A continual learning survey: Defying forgetting in classification tasks.

Cícero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying Relations by Ranking 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 (Volume 1: Long Papers)*, pages 626–634, Beijing, China.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

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, Online.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. FewRel: A Large-Scale Supervised Few-Shot Relation Classification Dataset with State-of-the-Art Evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4803–4809, Brussels, Belgium.

He He, Sheng Zha, and Haohan Wang. 2019. Unlearn dataset bias in natural language inference by fitting the residual. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 132–142, Hong Kong, China. Association for Computational Linguistics.

Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. 2020. AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty.

Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. 2020. End-to-end bias mitigation by modelling biases in corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8706–8716, Online. Association for Computational Linguistics.

Zipf George Kingsley. 1972. *Human behavior and the principle of least effort: An introduction to human ecology*. Hafner Publishing Company.

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*, 114(13):3521–3526.

Yuxuan Lai, Chen Zhang, Yansong Feng, Quzhe Huang, and Dongyan Zhao. 2021. Why machine reading comprehension models learn shortcuts? In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 989–1002, Online. Association for Computational Linguistics.

Hongtao Liu, Peiyi Wang, Fangzhao Wu, Pengfei Jiao, Wenjun Wang, Xing Xie, and Yueheng Sun. 2019. Reet: Joint relation extraction and entity typing via multi-task learning. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 327–339. Springer.

Arun Mallya and Svetlana Lazebnik. 2018. PackNet: Adding Multiple Tasks to a Single Network by Iterative Pruning. *arXiv:1711.05769 [cs]*.

Angrosh Mandya, Danushka Bollegala, and Frans Coenen. 2020. Contextualised Graph Attention for Improved Relation Extraction. *arXiv:2004.10624 [cs]*.

Matthias Minderer, Olivier Bachem, Neil Houlsby, and Michael Tschannen. 2020. Automatic shortcut removal for self-supervised representation learning. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Pramod Kaushik Mudrakarta, Ankur Taly, Mukund Sundararajan, and Kedar Dhamdhere. 2018. Did the model understand the question? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1896–1906, Melbourne, Australia. Association for Computational Linguistics.

Hao Peng, Tianyu Gao, Xu Han, Yankai Lin, Peng Li, Zhiyuan Liu, Maosong Sun, and Jie Zhou. 2020. 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)*, pages 3661–3672, Online.

Qi Qin, Wenpeng Hu, Han Peng, Dongyan Zhao, and Bing Liu. 2021. BNS: Building Network Structures Dynamically for Continual Learning. In *Advances in Neural Information Processing Systems*, volume 34, pages 20608–20620.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. 2017. iCaRL: Incremental Classifier and Representation Learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5533–5542, Honolulu, HI.

Joe Stacey, Pasquale Minervini, Haim Dubossarsky, Sebastian Riedel, and Tim Rocktäschel. 2020. Avoiding the Hypothesis-Only Bias in Natural Language Inference via Ensemble Adversarial Training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8281–8291, Online. Association for Computational Linguistics.

Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An empirical study on robustness to spurious correlations using pre-trained language models. *Transactions of the Association for Computational Linguistics*, 8:621–633.

Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C. Duchi, Vittorio Murino, and Silvio Savarese. 2018. Generalizing to unseen domains via adversarial data augmentation. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 5339–5349.

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, Volume 1 (Long and Short Papers)*, pages 796–806, Minneapolis, Minnesota.

Linlin Wang, Zhu Cao, Gerard de Melo, and Zhiyuan Liu. 2016. Relation Classification via Multi-Level Attention CNNs. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1298–1307, Berlin, Germany.

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. *arXiv preprint arXiv:2209.00243*.

Peiyi Wang, Runxin Xun, Tianyu Liu, Damai Dai, Baobao Chang, and Zhifang Sui. 2021. Behind the scenes: An exploration of trigger biases problem in few-shot event classification. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 1969–1978.

Tongtong Wu, Xuekai Li, Yuan-Fang Li, Reza Haffari, Guilin Qi, Yujin Zhu, and Guoqiang Xu. 2021. Curriculum-Meta Learning for Order-Robust Continual Relation Extraction. *arXiv:2101.01926 [cs]*.

Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. 2019. Large Scale Incremental Learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 374–382, Long Beach, CA, USA.

Minguang Xiao and Cong Liu. 2016. Semantic Relation Classification via Hierarchical Recurrent Neural Network with Attention. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1254–1263, Osaka, Japan.

Kun Xu, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2015. Semantic Relation Classification via Convolutional Neural Networks with Simple Negative Sampling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 536–540, Lisbon, Portugal.

Yan Xu, Ran Jia, Lili Mou, Ge Li, Yunchuan Chen, Yangyang Lu, and Zhi Jin. 2016. Improved Relation Classification by Deep Recurrent Neural Networks with Data Augmentation.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware Attention and Supervised Data Improve Slot Filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark.

Kang Zhao, Hua Xu, Jiangong Yang, and Kai Gao. 2022. Consistent Representation Learning for Continual Relation Extraction. *arXiv:2203.02721 [cs]*.

Long Zhao, Ting Liu, Xi Peng, and Dimitris Metaxas. 2020. Maximum-Entropy Adversarial Data Augmentation for Improved Generalization and Robustness. In *Advances in Neural Information Processing Systems*, volume 33, pages 14435–14447.

Fei Zhu, Zhen Cheng, Xu-Yao Zhang, and Chenglin Liu. 2021. Class-incremental learning via dual augmentation. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 14306–14318.

A Forgetting Rate on TACRED

We show the performance of two strong baselines on TACRED in Table 8. We also divide relations in TACRED into three groups according to their forgetting rate.

Model	ID	FR (%)	MS	F1	F1* (Δ)
EMAR	G1	2.2	0.49	93.5	95.7 (↑ 2.2)
	G2	7.1	0.64	76.5	85.8 (↑ 9.3)
	G3	13.7	0.75	56.9	68.0 (↑ 11.1)
RP-CRE	G1	2.3	0.49	93.2	95.7 (↑ 2.5)
	G2	7.5	0.62	76.8	86.0 (↑ 9.2)
	G3	15.3	0.76	56.2	67.8 (↑ 11.6)

Table 8: We divide relations of TACRED into 3 groups according to their forgetting rate (FR). "MS" is the short for *max similarity*. F1 and F1* are the Macro-F1 scores of EMAR/RP-CRE and the supervised model, respectively. Δ is the performance gap between 2 CRE models and the supervised model.

B Cases for performance curves of different relations

As illustrated in Figure 5, we provide cases to illustrate catastrophic forgetting only appears on some specific relations and significant performance degradation always occurs when analogous relation appears. For each relation, we plot the performance curves corresponding to five different task sequences. We notice that some relations almost have no performance degradation during the training process (as shown in the top row of Figure 5), while some relations suffer from catastrophic forgetting (as shown in the bottom three rows of Figure 5). We further observe that when the performance curve of a specific relation r has a sudden degradation, the corresponding task always contains relations very similar to r.

C Retrieval Test

As discussed in Section 4.2, a potential reason for catastrophic forgetting in CRE is the model only learns the spurious shortcuts in the continual learning setting. In order to evaluate the representation ability of the CRE model, we propose a retrieval test analysis.

Given an instance x, a CRE method utilizes an encoder f to encode its semantic features for learning and classifying relations,

$$\boldsymbol{h} = f(x). \tag{3}$$

For a relation, if its F1 score degrades greater than 0.1 after the model learning a new task, we consider it as an relation suffering severe forgetting. We group all relations suffering severe forgetting into a set R_f. For a relation $r \in R_f$, we additionally randomly sample 7 relations $\{r_1^*, ..., r_7^*\}$ (3 relations for TACRED) from $R \setminus R_f$ to build a pseudo task T^* containing instances from $R^* = \{r, r_1^*, ..., r_7^*\}$, where R is the relation set of the entire dataset. After training the CRE model on our built pseudo task T^*, we obtain the prototype $\boldsymbol{p}_r$ of the relation r, that is, the mean embedding of all instances belonging to r,

$$\boldsymbol{p}_r = \frac{\sum_{j=1}^{|r|} f(x_j^r)}{|r|}, \tag{4}$$

where $|r|$ is the number of instances of relation r. We also obtain embeddings of each instance in the entire test set,

$$I = \{\boldsymbol{h}_i | \boldsymbol{h}_i = f(x_i),\ x_i \in \bigcup_{r_j \in R} Q_j\}. \tag{5}$$

Then we compute the cosine similarity between $\boldsymbol{p}_r$ and $\boldsymbol{h}_i \in I$:

$$\text{Sim}(\boldsymbol{p}_r, \boldsymbol{h}_i) = \frac{\boldsymbol{p}_r \cdot \boldsymbol{h}_i}{|\boldsymbol{p}_r| \cdot |\boldsymbol{h}_i|}, \tag{6}$$

We consider rank-based metrics and use the mean precision at k ($P@k$), which is the proportion of instances whose label is r in the top-k similar set. Specifically, for FewRel, we use $P@100$ as metric. For TACRED, because this dataset has a severe imbalance problem and some relations only have less than 50 instances, we use mean $P@|Q_r|$ as metric, where $|Q_r|$ is the size of test set corresponding to relation r. If the retrieval precision is high, we can say that the model learns robust representations.

D Datasets

Following previous work (Han et al., 2020; Wu et al., 2021; Cui et al., 2021; Zhao et al., 2022), our experiments are conducted upon the following two widely datasets, and the training-test-validation split ratio is 3:1:1:

FewRel (Han et al., 2018) It is a relation extraction dataset originally proposed for few-shot learning, which contains 80 relations, each with 700 instances. Following Cui et al. (2021); Zhao et al. (2022), we use the training and validation set of FewRel for experimental.

TACRED (Zhang et al., 2017) It is a large-scale RE dataset built on news networks and online documents, containing 42 relations (including *no_relation*) and 106264 samples. Following Cui et al. (2021), *no_relation* was removed in our experiments, and the number of training samples for each relation is limited to 320 and the number of test samples of each relation to 40.

E Symmetric Relations in Two Datasets

In our reversed-class augmentation, we divide all relations into two categories, symmetric relation and asymmetric relation. The symmetric relation denotes the relation semantic is independent of which of the two given entities is the head or tail entity, and the relations except symmetric relations are asymmetric relations. (1) In FewRel, we find 2 symmetric relations, "P26 (spouse)" and "P3373 (sibling)". (2) In TACRED, we find 5 symmetric relations, "per:siblings", "org:alternate names", "per:spouse", "per:alternate names" and "per:other family".

F Robust Representation

In this section, we provide more cases to intuitively show the effectiveness of our look-ahead learning for learning robust representation. Please refer to Figure 6 for details.

G Relations in Different Groups

As discussed in Section 4, we divide all relations into three equal-sized groups based on their FR from small to large. In this section, we show example relations in Group 1 and Group 3 of FewRel in Table 9. It is easy to see that relations in Group 3, which suffer from significant performance degradation, have larger similarity to other relations in the dataset than relations in Group 1.

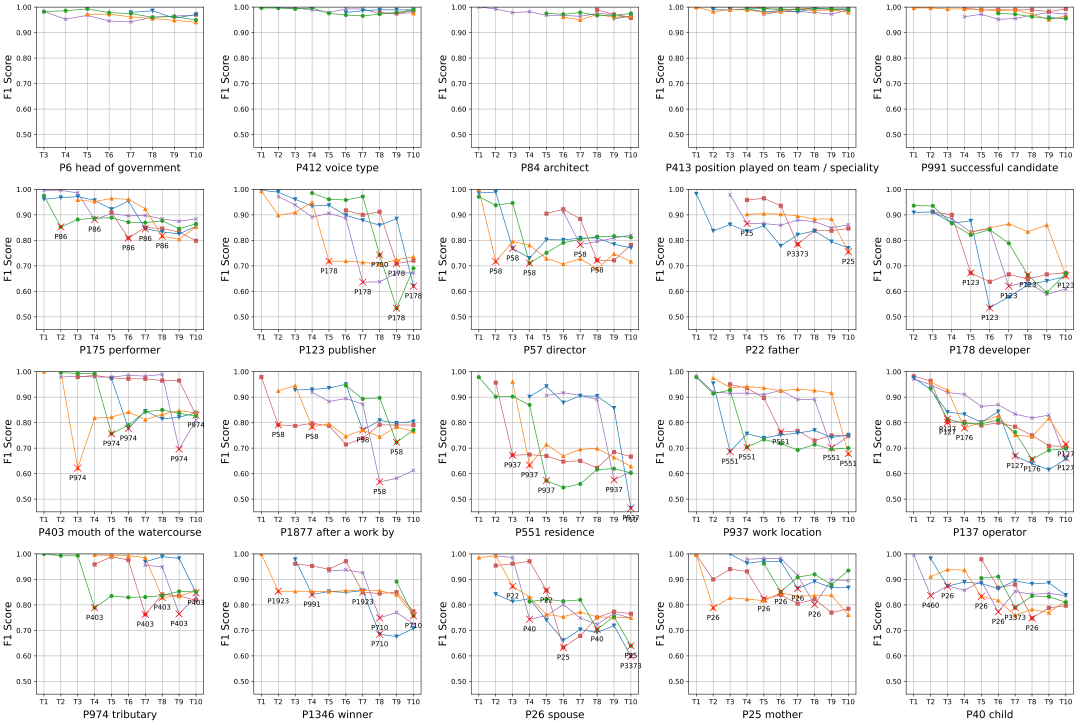


Figure 5: Cases of performance curves of different relations in FewRel. For each relation, we illustrate its F1 curves corresponding to five different task sequences. Note that the performance of some relations hardly degrades, while other relations suffer from catastrophic forgetting. For a specific relation r, we also plot bad cases (the F1 curve has a degradation more than 0.1 F1 score) containing top-5 most similar relations of r.

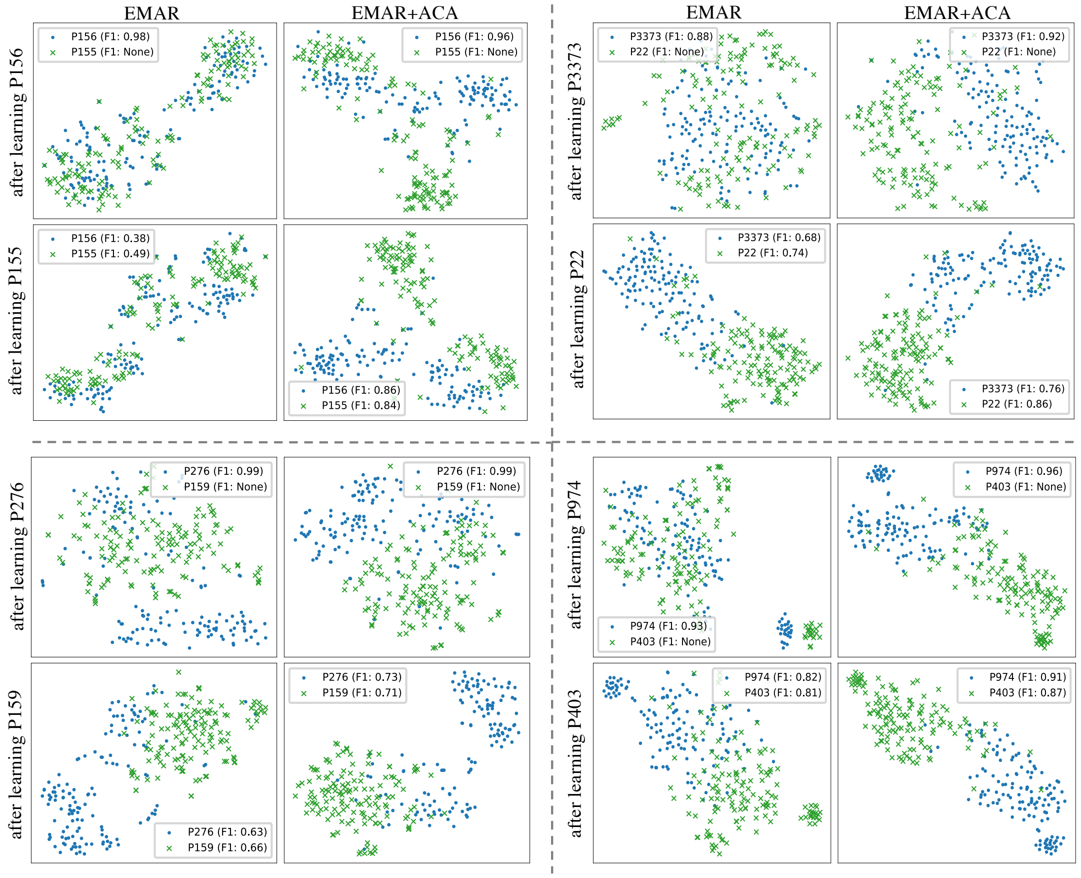


Figure 6: More cases of robust representation learning.

Example Relations of Group 1

Wikidata ID	Relation Name	Top-3 Most Similar Relations
P59	constellation	(0.693) P106, occupation (0.673) P27, country of citizenship (0.663) P641, sport
P1411	nominated for	(0.758) P449, original network (0.757) P750, distributor (0.755) P27, country of citizenship
P2094	competition class	(0.789) P641, sport (0.780) P463, member of (0.755) P27, country of citizenship
P105	taxon rank	(0.806) P31, instance of (0.794) P361, part of (0.784) P206, located in or next to body of water
P1435	heritage designation	(0.846) P177, crosses (0.823) P31, instance of (0.822) P131, located in the administrative territorial entity
P1344	participant of	(0.878) P27, country of citizenship (0.857) P463, member of (0.850) P551, residence
P410	military rank	(0.886) P39, position held (0.882) P241, military branch (0.848) P22, father
P84	architect	(0.890) P6, head of government (0.889) P127, owned by (0.875) P86, composer
P306	operating system	(0.891) P400, platform (0.881) P178, developer (0.856) P31, instance of
P1303	instrument	(0.894) P101, field of work (0.893) P463, member of (0.884) P106, occupation

Example Relations of Group 3

Wikidata ID	Relation Name	Top-3 Most Similar Relations
P155	follows	(0.991) P156, followed by (0.962) P361, part of (0.959) P527, has part
P706	located on terrain feature	(0.984) P206, located in or next to body of water (0.966) P131, located in the administrative territorial entity (0.961) P4552, mountain range
P57	director	(0.984) P58, screenwriter (0.974) P1877, after a work by (0.954) P86, composer
P22	father	(0.981) P40, child (0.974) P26, spouse (0.972) P3373, sibling
P123	publisher	(0.978) P178, developer (0.953) P750, distributor (0.943) P127, owned by
P127	owned by	(0.977) P355, subsidiary (0.967) P137, operator (0.958) P159, headquarters location
P25	mother	(0.976) P26, spouse (0.967) P40, child (0.961) P3373, sibling
P1877	after a work by	(0.974) P58, screenwriter (0.958) P57, director (0.931) P86, composer
P17	country	(0.968) P131, located in the administrative territorial entity (0.960) P361, part of (0.950) P159, headquarters location
P551	residence	(0.968) P937, work location (0.939) P27, country of citizenship (0.934) P159, headquarters location

Table 9: Example relations in Group 1 and Group 3 of FewRel. For each relation, we show its top-3 most similar relations with their corresponding cosine similarity.