Partial Domain Adaptation for Relation Extraction Based on Adversarial Learning

Xiaofei Cao, Juan Yang(✉), and Xiangbin Meng

Beijing Key Lab of Intelligent Telecommunication Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing, China
{caoxf,yangjuan,mengxb}@bupt.edu.cn

Abstract. Relation extraction methods based on domain adaptation have begun to be extensively applied in specific domains to alleviate the pressure of insufficient annotated corpus, which enables learning by utilizing the training data set of a related domain. However, the negative transfer may occur during the adaptive process due to differences in data distribution between domains. Besides, it is difficult to achieve a fine-grained alignment of relation category without fully mining the multi-mode data structure. Furthermore, as a common application scenario, partial domain adaptation (PDA) refers to domain adaptive behavior when the relation class set of a specific domain is a subset of the related domain. In this case, some outliers belonging to the related domain will reduce the performance of the model. To solve these problems, a novel model based on a multi-adversarial module for partial domain adaptation (MAPDA) is proposed in this study. We design a weight mechanism to mitigate the impact of noise samples and outlier categories, and embed several adversarial networks to realize various category alignments between domains. Experimental results demonstrate that our proposed model significantly improves the state-of-the-art performance of relation extraction implemented in domain adaptation.

Keywords: Relation extraction · Domain adaptation · Adversarial learning

1 Introduction

Relation extraction (RE) plays a pivotal role in addressing the issue of information extraction, which aims to detect the semantic relationship between real-world entities. For instance, the task of RE can be described as discovering the "cause-effect (e1, e2)" relation between a pair of entities <*microphone, signal*> in the sentence: the microphone converts sound into an electrical signal. RE has been widely utilized in various fields of natural language processing (NLP), such as automatic question and answering system [1] and knowledge graphs (KG) [2,3]. The semantic web is a general framework proposed to make the data on the network machine-readable [4], and which utilizes the resource description

A. Harth et al. (Eds.): ESWC 2020, LNCS 12123, pp. 89–104, 2020.
https://doi.org/10.1007/978-3-030-49461-2_6

framework (RDF) to describe network resources. The edge element of RDF represents the relation between entities or the relationship between the entity and its attributes. Therefore, RE indirectly provides data support for the construction of the semantic network.

Extensive research has demonstrated that RE models based on deep learning indicate outstanding performance with a large quantity of corpus. Zeng et al. [5] applied the convolution neural network (CNN) to automatically gain lexical and sentence features. Socher et al. [6] proposed using the recurrent neural network (RNN) to explore the combinatorial vector representation of phrases and sentences of any syntactic type and length. These models based on deep learning can automatically learn the implicit and complex feature expression of text. Therefore, they are considered to be better than those based on traditional machine learning algorithms such as SVM [7] and MaxEnt [8]. However, in some domains, the lack of sufficient annotation data set for model training can lead to poor performance. In order to relieve the pressure of labeled data sparsity, Mintz et al. [9] presented distant supervision (DS). DS takes the triple <e1, r, e2> in the existing knowledge base as the seed. It then matches the text containing e1 and e2 heuristically, and the resulting sentences are used as the annotation data of the r relationship. However, this method will generate much noise. For example, triple <Donald Trump, born in, New York>, may be aligned to "Donald Trump was born in New York", or may be aligned to "Donald Trump worked in New York". The first one is the annotation data that we want to generate, while the second one is the noise data. How to remove the noise data is an important research topic, which to date has had limited exploration. To complicate things further, the precondition of DS is dependent up the existence and quality of the knowledge base.

Pan et al. [10] found that domain adaptation (DA) can assist a target domain training model by using annotation data of the source domain. It has been widely used in computer vision, NLP, and other related fields. For example, predicting the emotion of data generated from the fast-food comment is done by utilizing movie comment data using existing emotional markers [11], or classified picture data on the e-commerce website is used to classify photos taken by mobile phones [12]. By eliminating the limitation that training data and test data must be independent and equally distributed, DA provides an effective way for RE to be applied in a data-sparse domain. Plank et al. [13] combined term generalization approaches and structured kernels to improve the performance of a relation extractor on new domains. Nguyen et al. [14] evaluated embedding words and clustering on adapting feature-based relation extraction systems. All of these research studies were done to find a way to effectively improve the model accuracy on new domains through DA. However, we discover additional problems in DA that required further resolutions.

– **Model collapse.** Model collapse refers to when most DA models focus on reducing the feature-level domain shift, even in the same feature space, category mismatch problem may exist and result in poor migration to the new dataset [15]. For example, some entity pairs are assigned the wrong relation

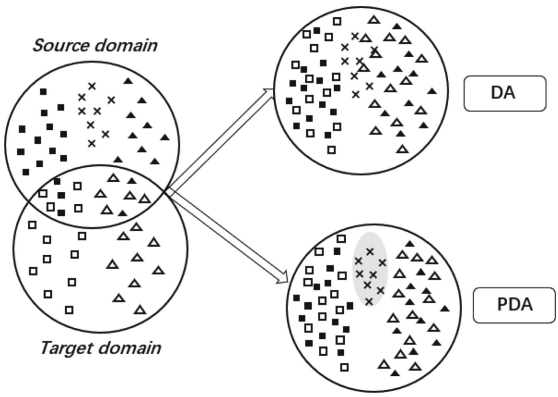

Fig. 1. DA represents the general domain adaptation. PDA is a generalized domain adaptive setting where the classification space of the target domain is a subset of the source domain category space. The red mark x in the figure represents the outlier class. It only appears in the source domain data, which may lead to a negative transfer.

types, which demonstrates that the model lacks robustness. Consequently, a more fine-grained class alignment solution needs to be developed.

- **Outlier classes.** Current DA models are generally based on the assumption that the source domain and target domain share the same category space. However, the PDA usually exists where the class set of target domain is a subset of the source domain. For example, a general domain such as Wikipedia partially adapts to a vertical domain (such as news domain or financial domain) with smaller label space. In this case, outlier classes that only belong to the source domain may lead to the reduction of the classification effect of the source supervised model [16].
- **Negative migration samples.** Because the source and target domain differ at the feature level, there may be some non-migratable samples. If such samples of the source domain are fitted to align with the samples of the target domain, it can negatively affect model performance. It is therefore considered important to determine how to reduce the impact of these samples on the network during migration. This is one of the key issues that need to be resolved to improve the accuracy of the model.

To address the above problems, we work on ways to alleviate negative transfer via the PDA solution with a weight selection mechanism. This approach is expected to reduce negative migration and improve the generalization abilities of the model. As shown in Fig. 1, the ellipse consisting of crosses in the middle of the circle has been separated to limit the migration of the outliers. We subsequently strive to align the labels of source and target domains by embedding multiple adversarial neural networks, aiming to eliminate the hidden dangers of category mismatches.

In summary, we propose a novel RE model to address the aforementioned problems by utilizing a weight mechanism to reduce the impact of negative transfer. This approach is based on adversarial learning to achieve the alignment of categories between different domains. Furthermore, our study provides new insights and understanding into partial domain adaptation learning of RE. As far as we have been able to determine, our model is the first one to apply a multi-layer adversarial network of RE. The results of our experimental study demonstrate that compared with other baseline models, our model is able to consistently achieve state-of-the-art performance for various partial domain adaptation tasks.

2 Related Work

2.1 Relation Extraction

In recent years, the area of DS has received significant research attention. This research was presented to combine the advantages of bootstrapping [17] and supervised learning, to alleviate the pressure of missing training data sets. Subsequent DS research focused on two key aspects. Many classic models have enhanced the robustness of the RE model by reducing the training weight of the noisy sample. In order to solve the problem of error tagging in DS, Zeng et al. [18] proposed a multi-instance learning method to extract a high confidence training corpus for a RE model. Liu et al. [19] introduced a sentence level attention mechanism into multiple-instance learning, which has effectively reduced the weight of noise instances. However, multi-instance learning is based on the assumption that there is at least one correct labeled data in each package. Luo et al. [20] suggested using a noise matrix to fit with the distribution of noise, so as to achieve the purpose of fitting it with the real distribution. Several other models tried to improve the accuracy of the RE model by taking full advantage of the syntactic information. Zhang et al. [21] supported the notion that encoding the main words on the dependency path of sentences by a network block GRU could capture more important information in sentences. Liu et al. [22] applied bidirectional gated recurrent unit to extract sentence feature vectors from each word, and an attention mechanism to give greater weight to keywords.

However, all of these models required sufficient labeling data or prior knowledge to build fake samples, which ignored relevant information in other related domains. Our model focuses on the adaptive learning of RE, which removes restrictions of prior knowledge, to transfer the knowledge acquired by the supervised model of a general domain to a special field.

2.2 Adversarial Domain Adaptation

The research study [25] first proposed the idea of adversarial domain adaptation to embed domain adaption into the process of learning representation, so that the decision making about final classifications thoroughly integrated the characteristics of differences and variances to the domain change. In this way, the

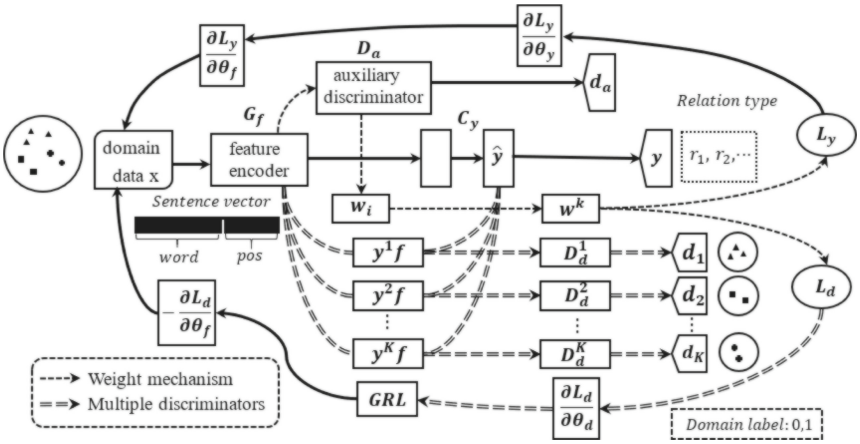

Fig. 2. The architecture of our method. The G_f denotes the feature extractor CNN to capture the text information, and the C_y represents the relation classifier. The auxiliary discriminator D_a is the core structure of the weight mechanism, which is introduced to obtain the sample weight w_i and iteratively updates category weight w^k that is attached to the loss function of the discriminator and the classifier. Besides, K discriminators are applied to capture a multi-mode data structure [23]. For example, the k-th discriminator is denoted as D_d^k. A gradient reversal layer (GRL) [24] is used to illustrate the opposite value of the gradient and achieve the effect of confrontation.

feedforward network could be applied to a new domain without being affected by the displacement influences between the two domains. Subsequently, research studies on adversarial domain adaptation have emerged. Among them, a few papers have drawn attention to the negative effects of transfer [26] and the risk of model collapse [27,28]. One of these papers [23] presented a structure of multiple discriminators to explore the multi-mode structure, while it ignored PDA. Cao et al. [16] weighted the data of the anomaly source class to train the source classifier and to promote positive delivery by matching the feature distribution in the shared label space. Cao et al. [29] found a suitable solution by decreasing the weight of the noise sample or outlier to update network parameters.

However, the research and application direction of DA methods based on adversarial network have mainly focused on the image domain to conduct the image classification [30,31]. There has been a lack of systematic discussion and research work in the field of relation extraction. Plank and Moschitti [13] found that a proper combination of grammar and lexical generalization was useful for DA. Zhang et al. [32] proposed a novel model of relation-gated adversarial learning for relation extraction to extend the adversarial based DA methods. However, this approach may cause problems in that even if the network training converged and the discriminator was completely confused, it would be impossible to tell which domain the sample came from. There was no guarantee that the shared feature distribution of data could be captured.

3 Methodology

3.1 Problem Definition

Given the labeled source domain data set $D_s = (x_i, y_i)_{i=1}^{n_s}$ with $|C_s|$ categories and the unlabeled target domain data set $D_t = (x_i)_{i=1}^{n_t}$ with $|C_t|$ categories. We assume that $|C_s| >> |C_t|$. The goal of this research is to design an adversarial neural network that captures transferable information $f = G_f(x)$ and the adaptive classifier C_y. This section will illustrate in detail, including the mechanisms and implementation of the model. The model structure is shown in Fig. 2.

3.2 Feature Extractor

A feature extractor is used to get the text features in the source and target domains. From this aspect, there are many effective supervision models and network structures, such as CNN [5], Bi-LSTM [33], and PCNN [18]. This paper adopts a CNN structure, which extracts features by concatenating lexical features and sentence features. For input text sample x_i, its semantic features are expressed as $f = G_f(x_i)$. G_f is the symbolic representation of CNN. By giving the characteristics of source domain samples to the C_y classifier, the probability of each relational class and the prediction label can be obtained. The following loss function is established to update the parameters of the classifier and the encoder.

$$(\hat{\theta_f}, \hat{\theta_y}) = \underset{\theta_f, \theta_y}{argmin} \frac{1}{n_s} \sum_{x_i \in D_s} L_y(C_y(G_f(x_i)), y_i) \tag{1}$$

In the above formula, θ_f is the parameter of CNN, θ_y is the parameter of the classifier, and y_i is the true label of sample x_i. L_y adopts a cross-entropy loss function.

3.3 Multi-adversarial Neural Network

The core idea of the adversarial domain adaptation is inspired by generative adversarial networks [34], which consists of a generator and a discriminator. The generator randomly takes samples from the source domain as input, and its output results should imitate the real samples in the target domain. The discriminator takes the real sample of the target domain or the output of the generator as the input. It is designed to focus on distinguishing the output of the generator from the actual sample to the greatest extent, while the generator should cheat the discriminator as far as possible. The two networks constitute an adversarial neural network, confronting each other and continuously adjusting the parameters. The ultimate goal of the adversarial neural network is to make the discriminator unable to judge whether the output of the generator is the target domain sample. This approach can maintain the feature invariance between the source domain and reduce the discrepancy of data distribution.

In this study, the feature extractor acts as a generator, and we use the symbol G_d to represent the discriminator. The symbol L_d denotes the optimizer goal of the adversarial neural network, which can be expressed as follows.

$$\min_{\theta_f} \max_{\theta_d} L_d(\theta_d, \theta_f) = \int_{x_s} p(x_s) log G_d(x_s) dx_s + \int_{x_t} p(x_t) log G_d(G_f(x_t)) dx_t \quad (2)$$

The $p(x_s)$ denotes data distribution in the source domain and the $p(x_t)$ represents data distribution in the target domain, noting that $p(x_s) \neq p(x_t)$. The objective of the above optimization function is to align two distributions, $p(x_s)$ and $p(x_t)$.

However, these strategies are far from enough to improve the performance of the RE model in the target domain. From an existing defect, a single domain discriminator does not take advantage of the complex multi-mode structure. Consequently, in this paper, a multi-adversarial domain adaptive (MADA) module [23] is applied to capture the multi-mode structure to ensure the fine-grained alignment of different data distributions.

Assuming that there are K classes in the source domain, the model uses K discriminators, with each discriminator focusing on aligning a certain cross-class in the source domain and the target domain. The optimized objective function of the discriminator is as follows:

$$L_d = \frac{1}{n_s + n_t} \sum_{k=1}^{K} \sum_{x_i \in D} L_d^k (D_d^k (\hat{y_i^k} G_f(x_i)), d_i)) \quad (3)$$

The overall objective function can be expressed as the following formula:

$$L(\theta_f, \theta_y, \theta_d|_{k=1}^K) = \frac{1}{n_s} \sum_{x_i \in D_s} L_y(C_y(G_f(x_i)), y_i) \quad (4)$$

$$-\frac{\lambda}{n_s + n_t} \sum_{k=1}^{K} \sum_{x_i \in D} L_d^k (D_d^k (\hat{y_i^k} G_f(x_i)), d_i)$$

Where θ_d^k is the parameter of D_d^k, L_d^k denotes the loss function of the k-th discriminator, and $\hat{y_i^k}$ represents the probability that the sample x_i belongs to class k. In addition, $D = D_s \cup D_t$. The first part of the formula represents the loss function of the relation classifier, while the second part represents the loss function of the K discriminators.

3.4 Adaptive Transfer Weight Selection Mechanism

DA is not expected to the situation of $c \in C_s$ and $c \notin C_t$. The previous network structures saw, the samples of each category in the source domain fitted with target domain data without differences, which was not conducive to the model performance of the target domain. In this paper, the weight mechanism is utilized to control the loss function to mitigate the migration of the negative samples and enhance the adaptability of the positive samples.

Instance Weight Calculating. The sample migration ability can be reflected in the discriminator's prediction of the probability that the sample originated from the source domain. The higher the predicted confidence, the more likely the sample can be distinguished from the target domain sample [26]. On the contrary, if the sample has low predicted confidence, this can suggest that the source domain sample and the target domain sample have a higher similarity. At this stage, the source domain sample has more migration performance, which means that the model needs to increase to fit with the sample. The migration weight therefore can be set by the output of the discriminator so that by using the source domain sample as the input of the classification model, the migration weight can be set according to the migration performance.

In this paper, we are able to improve the influence of the sample with low prediction confidence on neural network parameters. Specifically, an auxiliary discriminator D_a is introduced into the model, and the sample weight is constructed by predicting the result of the auxiliary discriminator. The higher the confidence, the greater the weight. Otherwise, the weight will be smaller. The prediction confidence of the sample is denoted as $D_a(f)$ and its weight w_i can be calculated by using the following formula:

$$w_i = \frac{1}{1 + \frac{D_a(G_f(x_s))}{D_a(G_f(x_t))}} = 1 - D_a(f) \tag{5}$$

Class Weight Updating. In order to resolve the central problem of negative transfer caused by outlier categories, the uncertainty of sample migration is used to calculate the category weight. Obviously, all of the samples in an outlier class should not have the nature of migration, so the mobility of the samples can measure the mobility of the category of relation to a certain extent. If all samples in a relation class have low mobility, the class mobility should also be relatively low. Therefore, the migration weight of the class can be calculated by samples weights, so as to reduce the migration weight of the outlier categories.

The larger the w^k is, the closer the class is to the target domain category. Otherwise, there is a greater probability of it being considered an outlier. The effect of category weight on the model is reflected in the following aspects: it strengthens the influence of the category weight on the relation classifier; or, the influences of the samples in the source domain on the discriminator and feature extractor parameters are enhanced. The formula for calculating the category

weight is expressed as $\frac{1}{n_{sk}} \sum_{i=1}^{n_{sk}} w_i$. According to the weights of classes, the influence of outliers on parameter updating is effectively limited. The w^K is initialized to $w^K = [1, 1, \cdots, 1]$. Obviously, for outliers, the migration of the interference samples is finite.

Table 1. ACE05 entity types and relation types.

Entity types	Relation types
FAC (Facility)	ART (artifact)
GPE (Geo-Political Entity)	GEN-AFF (Gen-affiliation)
LOC (Location)	ORG-AFF (Org-affiliation)
ORG (Organization)	PART-WHOLE (part-whole)
PER (Person)	PER_SOC (person-social)
VEH (Vehicle)	PHYS (physical)
WEA (Weapon)	——

3.5 Loss Function

The following formula represents the total loss function of our model. The w_i^k represents the migration weight of the category to which the sample x_i belongs. The first part is the loss of a training relation classifier with the source domain data. It emphasizes the use of samples from high mobility categories to update the classification model parameters, which can enhance the generalization performance of the supervised model in the target domain. The second part is the discriminator loss function of K discriminators. On the one hand, w_i^k avoids assigning each sample point to only one discriminator. On the other hand, each sample point is only aligned with the most relevant class, and the uncorrelated class is filtered out by probability. It is not included in the corresponding domain discriminator, thus avoiding the wrong alignment of the discrimination structure in different distributions. With the updating of the class weight, the probability of outliers will gradually converge. In addition, the impact on parameter updating of the discriminators and feature extractor will reduce.

$$L(\theta_f, \theta_y, \theta_d^k|_{k=1}^K) = \frac{1}{n_s} \sum_{k=1}^{K} \sum_{x_i \in D_s} w_i^k L_y(C_y(G_f(x_i)), y_i) \tag{6}$$

$$- \lambda \sum_{k=1}^{K} (\sum_{x_i \in D_t} L_d^k(D_d^k(\hat{y_i^k} G_f(x_i)), d_i) + \sum_{x_i \in D_s} w_i^k L_d^k(D_d^k(\hat{y_i^k} G_f(x_i)), d_i))$$

The optimal parameters of the model are expressed as follows.

$$\begin{aligned} (\hat{\theta_f}, \hat{\theta_y}) &= \underset{\theta_f, \theta_y}{argmin}\, L(\theta_f, \theta_y, \theta_d^k|_{k=1}^K), \\ (\hat{\theta}_d^1, \cdots, \hat{\theta}_d^K) &= \underset{\theta_d^1, \cdots, \theta_d^k}{argmax}\, L(\theta_f, \theta_y, \theta_d^k|_{k=1}^K) \end{aligned} \tag{7}$$

4 Experiments

4.1 Dataset

ACE05 Dataset. ACE05 corpus is a type of data that is released by linguistic data consortium. It consists of entities, relations, and event annotations. It aims at developing automatic content extraction technology, and it supports automatic processing of human language in the form of text. This data set includes seven types of entities and six types of relations (see Table 1). In this study, we used the ACE05 dataset to evaluate our proposed model by dividing its texts from its six genres into domains: broadcast conversation (bc), broadcast news (bn), telephone conversation (cts), newswire (nw), usenet (un) and weblogs (wl). To get an understanding of how these domains differ, Fig. 3 depicts the distribution of relations in each domain.

NYT-10 Dataset. NYT-10 dataset has been extensively used in DS research, which was originally developed by Riedel et al. [35], and it was generated by aligning Freebase relations with the New York Times (NYT) corpus. Entity mentions are determined using the Stanford named entity tagger [36], and they are further matched to the names of Freebase entities. This corpus includes 52 relations and a special relation NA which means that there is no relation between the entity pair in this instance. NYT-10 corpus is composed of training data and testing data, where data from 2005–2006 are used as the training set, and data from 2007 is used for testing. Training data includes 522,611 sentences, 281,270 entity pairs, and 18,252 relational facts. Testing data includes 172,448 sentences, 96,678 entity pairs and 1,950 relational facts. We evaluate the performance of our model under an setting using this dataset.

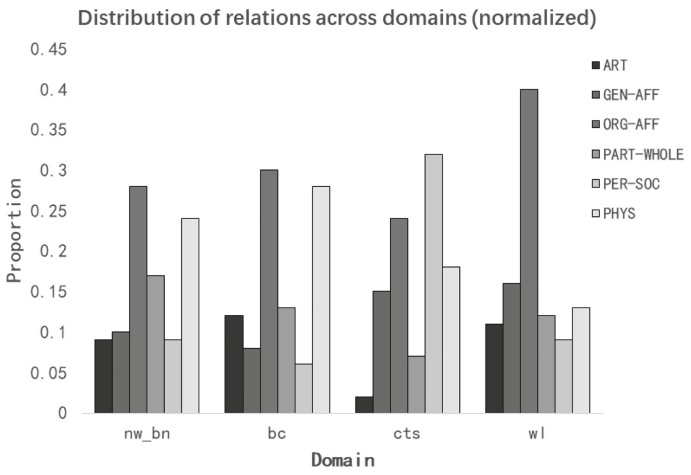

Fig. 3. Distributions of relations in ACE05.

4.2 Hyperparameters Settings

In order to fairly compare the results of our models with those baselines, we follow most of the experimental parameters in existing research [37], which proposed an unsupervised domain adaptation model consisting of a CNN-based relation classifier and a domain-adversarial classifier. We use word embedding that is pre-trained on newswire with 300 dimensions from word2vec [38] as the input of the CNN model. We also choose a cross-validation approach to tune our model and conduct a grid search to determine model parameters.

Table 2. Partition of ACE05 data set and overview of corpus.

Split	Corpus	Documents	Sentences	ASL	Relations
Source domain	nw & bn	298	5029	18.8	3562
Target domain	bc	52	2267	16.3	1297
	wl	114	1697	22.6	677
	cts	34	2696	15.3	603

4.3 Evaluation Results

Results on ACE05. In terms of data set division, previous works [32,37] used newswire (bn & nw) as the source data. The other half of bc, cts, and wl were the target training data, and the other half of bc, cts, and wl as the target test data. We use the same data split process (see Table 2). Our model require unlabeled target domain instances. To meet this requirement and avoid the train-on-test, for all of the three test domains, we separate 20% of the data from the training set as a validation set, in order to adjust the hyperparameters in the model. In terms of experimental settings, several experiments are set up to compare our proposed model with existing models. We choose to design two directions for our comparison. On the one hand, we set a conventional domain adaptation, which extracts some of the relational categories from C_s to make $C_s = C_t$. Reference experiments are as follows: Hybrid [39] combined the traditional feature-based methods, CNN and RNN, and the FCM was used for compositional embedding. CNN+DANN [37] contained a CNN-based relational classifier and a domain-adversarial classifier. CNN+MADA has been modified on the basis of the prototype, replacing the original feature extraction model with a CNN structure. Other parts of the model have not been altered. MADA-weight was designed on the basis of CNN+MADA. The weight mechanism was only valid for the loss function of the classifier and does not affect the loss function of the discriminator.

On the other hand, we promote the adaptive comparison of partial domains. The relational category sets of the three target domains have the following associations, which is to guarantee that $C_t \neq C_s$ and $C_t \in C_s$. CNN + DANN is used as the baseline model to compare with our final MAPDA model.

The experimental results are shown in the following table. The bold word in the table represents that the F1 score of the model has improved compared

Table 3. Comparisons with classical models on F1-score in two aspects: formal domain adaptation and partial domain adaptation. Bold font represents the corresponding model effect, which has demonstrated distinct improvements.

Normal DA	bc	wc	cts	Avg
FCM	61.90	N/A	N/A	N/A
Hybrid	63.26	N/A	N/A	N/A
CNN+DANN	65.16	55.55	57.19	59.30
CNN+MADA	64.23	54.36	55.28	57.96
MADA-weight	**65.86**	**56.10**	56.33	**59.43**
Partial DA	bc	wl	cts	Avg
CNN+DANN	63.17	53.55	53.32	56.68
MAPDA	**65.71**	**56.01**	**55.12**	**59.03**

Table 4. Comparisons of different methods under domain adaptive and non domain adaptive settings.

No DA	Top 100	Top 200	Top 300	Avg
CNN	0.62	0.60	0.60	0.61
PCNN	0.66	0.63	0.62	0.64
DA	Top 100	Top 200	Top 300	Avg
CNN	0.85	0.80	0.76	0.80
PCNN	0.87	0.84	0.82	0.84
CNN+DANN	0.80	0.75	0.71	0.75
MADA-weight	**0.87**	**0.86**	**0.83**	**0.85**

with other models. From the evaluation results that are shown in Table 3, the following points can be observed and summarized. Firstly, in the case of normal DA, the performance of applying MADA directly to relation extraction need to be improved. Our model MADA-weight achieves a performance comparable to that of CNN+DANN, which is a recognized state-of-the-art model. The model demonstrates that it is an effective option to apply sample weight and category weight to the loss function of a classification supervision model, and alleviate the migration of negative samples. Secondly, in the case of partial DA, our model significantly outperforms the plain adversarial DA model. These positive results demonstrate the validity of our weight mechanism and the multi-adversarial adaptive layer.

Results on NYT-10. In our experiment, we take samples of prediction probability Top N (N is 100, 200, 300 respectively), and ignore NA class. We then use the prediction results of this part of the data to evaluate the model performance. The results of the evaluation on NYT-10 can be seen in Table 4. We set

up two comparative experiments. One is an experiment without domain adaptive method (No DA), including CNN and PCNN models. In this setting, after training with source domain data, the model is directly applied to the samples in the target domain for prediction. The other experiment use the adaptive domain method (DA), including CNN, PCNN, CNN + DANN, and our model MADA-weight. The models use the source domain data for training, and we then apply the labeled data of the target domain for either fine-tuning or by applying the adaptive domain method for transfer learning.

From the results of the experiment, we can see that the accuracy of the CNN and PCNN models without DA is stable between 0.6 and 0.7. The highest accuracy is 0.66, while CNN and PCNN with DA are found to be 0.8 and above. These results demonstrate that DA is effective in an unsupervised environment and has a positive role in improving the accuracy of the RE model.

Furthermore, in the setting of the top 100, our model MADA with a weight mechanism (MADA-weight) gains 0.87 and exceeds other models by an average of 0.85. It achieved an optimal effect compared with other DA methods in the DA column, which further demonstrates that our weight mechanism is effective.

5 Conclusion

In this study, we propose a novel model based on adversarial learning to extract relation, which successfully obtains an improvement on all three test domains of ACE05 in partial domain adaptation. In addition, the results are able to demonstrate the practicability of the weight mechanism on the NYT-10 dataset. We use multiple adversarial neural networks to learn cross-domain features and align data distribution of the source domain and target domain. It will be a useful instrument for RE to relieve the pressure of data sparsity. Future studies will focus on the scenario where the set of relational categories for the source and target domains only partially overlap. We believe that this research will have a considerable impact on the outcomes, reflects an extensive application value, and generate new research studies in this field.

References

1. Cabrio, E., Cojan, J., Aprosio, A.P., Magnini, B., Lavelli, A., Gandon, F.: QAKIS: an open domain QA system based on relational patterns. In: Proceedings of the 2012 International Conference on Posters and Demonstrations Track-Volume 914 (2012)
2. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: Twenty-Ninth AAAI Conference on Artificial Intelligence (2015)
3. Schlichtkrull, M., Kipf, T.N., Bloem, P., van den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: Gangemi, A., et al. (eds.) ESWC 2018. LNCS, vol. 10843, pp. 593–607. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93417-4_38

4. Gutierrez, C., Hurtado, C.A., Mendelzon, A.O., Pérez, J.: Foundations of semantic web databases. J. Comput. Syst. Sci. **77**(3), 520–541 (2011)
5. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J., et al.: Relation classification via convolutional deep neural network. In: Proceedings of the 25th International Conference on Computational Linguistics (2014)
6. Socher, R., Huval, B., Manning, C.D., Ng, A.Y.: Semantic compositionality through recursive matrix-vector spaces. In: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1201–1211. Association for Computational Linguistics (2012)
7. GuoDong, Z., Jian, S., Jie, Z., Min, Z.: Exploring various knowledge in relation extraction. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 427–434. Association for Computational Linguistics (2005)
8. Kambhatla, N.: Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In: Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions, p. 22-es (2004)
9. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2, vol. 2, pp. 1003–1011. Association for Computational Linguistics (2009)
10. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2009)
11. Glorot, X., Bordes, A., Bengio, Y.: Domain adaptation for large-scale sentiment classification: a deep learning approach. In: ICML (2011)
12. Gebru, T., Hoffman, J., Fei-Fei, L.: Fine-grained recognition in the wild: a multi-task domain adaptation approach. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1349–1358 (2017)
13. Plank, B., Moschitti, A.: Embedding semantic similarity in tree kernels for domain adaptation of relation extraction. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1498–1507 (2013)
14. Nguyen, T.H., Grishman, R.: Employing word representations and regularization for domain adaptation of relation extraction. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 68–74 (2014)
15. Long, M., Cao, Z., Wang, J., Jordan, M.I.: Conditional adversarial domain adaptation. In: Advances in Neural Information Processing Systems, pp. 1640–1650 (2018)
16. Cao, Z., Ma, L., Long, M., Wang, J.: Partial adversarial domain adaptation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11212, pp. 139–155. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01237-3_9
17. Brin, S.: Extracting patterns and relations from the world wide web. In: Atzeni, P., Mendelzon, A., Mecca, G. (eds.) WebDB 1998. LNCS, vol. 1590, pp. 172–183. Springer, Heidelberg (1999). https://doi.org/10.1007/10704656_11
18. Zeng, D., Liu, K., Chen, Y., Zhao, J.: Distant supervision for relation extraction via piecewise convolutional neural networks. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1753–1762 (2015)

19. Lin, Y., Shen, S., Liu, Z., Luan, H., Sun, M.: Neural relation extraction with selective attention over instances. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 2124–2133 (2016)
20. Luo, B., et al.: Learning with noise: enhance distantly supervised relation extraction with dynamic transition matrix. arXiv preprint arXiv:1705.03995 (2017)
21. Zhang, Y., Qi, P., Manning, C.D.: Graph convolution over pruned dependency trees improves relation extraction. arXiv preprint arXiv:1809.10185 (2018)
22. Liu, T., Zhang, X., Zhou, W., Jia, W.: Neural relation extraction via inner-sentence noise reduction and transfer learning. arXiv preprint arXiv:1808.06738 (2018)
23. Pei, Z., Cao, Z., Long, M., Wang, J.: Multi-adversarial domain adaptation. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
24. Ganin, Y., et al.: Domain-adversarial training of neural networks. J. Mach. Learn. Res. **17**(1), 2030–2096 (2016)
25. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. arXiv preprint arXiv:1409.7495 (2014)
26. Zhang, J., Ding, Z., Li, W., Ogunbona, P.: Importance weighted adversarial nets for partial domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8156–8164 (2018)
27. Luo, Y., Zheng, L., Guan, T., Yu, J., Yang, Y.: Taking a closer look at domain shift: category-level adversaries for semantics consistent domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2507–2516 (2019)
28. Zhang, Y., Tang, H., Jia, K., Tan, M.: Domain-symmetric networks for adversarial domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5031–5040 (2019)
29. Cao, Z., You, K., Long, M., Wang, J., Yang, Q.: Learning to transfer examples for partial domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2985–2994 (2019)
30. Xie, S., Zheng, Z., Chen, L., Chen, C.: Learning semantic representations for unsupervised domain adaptation. In: International Conference on Machine Learning, pp. 5423–5432 (2018)
31. Pan, Y., Yao, T., Li, Y., Wang, Y., Ngo, C.W., Mei, T.: Transferrable prototypical networks for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2239–2247 (2019)
32. Zhang, N., Deng, S., Sun, Z., Chen, J., Zhang, W., Chen, H.: Transfer learning for relation extraction via relation-gated adversarial learning. arXiv preprint arXiv:1908.08507 (2019)
33. Miwa, M., Bansal, M.: End-to-end relation extraction using LSTMs on sequences and tree structures. arXiv preprint arXiv:1601.00770 (2016)
34. Goodfellow, I., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
35. Riedel, S., Yao, L., McCallum, A.: Modeling relations and their mentions without labeled text. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) ECML PKDD 2010. LNCS (LNAI), vol. 6323, pp. 148–163. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15939-8_10
36. Finkel, J.R., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by Gibbs sampling. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 363–370. Association for Computational Linguistics (2005)

37. Fu, L., Nguyen, T.H., Min, B., Grishman, R.: Domain adaptation for relation extraction with domain adversarial neural network. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pp. 425–429 (2017)
38. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
39. Nguyen, T.H., Grishman, R.: Combining neural networks and log-linear models to improve relation extraction. arXiv preprint arXiv:1511.05926 (2015)