Information Sciences 573 (2021) 222–238

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Classifier-adaptation knowledge distillation framework for relation extraction and event detection with imbalanced data

Dandan Song *, Jing Xu, Jinhui Pang, Heyan Huang

School of Computer Science and Technology, Beijing Institute of Technology, China

ARTICLE INFO

Article history:
Received 16 November 2020
Received in revised form 12 May 2021
Accepted 19 May 2021
Available online 26 May 2021

Keywords:
Data imbalance
Identification information
Knowledge distillation
Relation extraction
Event detection

ABSTRACT

Fundamental information extraction tasks, such as relation extraction and event detection, suffer from a data imbalance problem. To alleviate this problem, existing methods rely mostly on well-designed loss functions to reduce the negative influence of imbalanced data. However, this approach requires additional hyper-parameters and limits scalability. Furthermore, these methods can only benefit specific tasks and do not provide a unified framework across relation extraction and event detection. In this paper, a Classifier-Adaptation Knowledge Distillation (CAKD) framework is proposed to address these issues, thus improving relation extraction and event detection performance. The first step is to exploit sentence-level identification information across relation extraction and event detection, which can reduce identification errors caused by the data imbalance problem without relying on additional hyper-parameters. Moreover, this sentence-level identification information is used by a teacher network to guide the baseline model's training by sharing its classifier. Like an instructor, the classifier improves the baseline model's ability to extract this sentence-level identification information from raw texts, thus benefiting overall performance. Experiments were conducted on both relation extraction and event detection using the Text Analysis Conference Relation Extraction Dataset (TACRED) and Automatic Content Extraction (ACE) 2005 English datasets, respectively. The results demonstrate the effectiveness of the proposed framework.

1. Introduction

Information extraction, which aims to capture structural information from plain texts, plays an important role in many downstream tasks, including reading comprehension, automatic text summarization, question answering, knowledge graph augmentation, and others. Among information extraction tasks, relation extraction and event detection are two of the most vital. Specifically, relation extraction discriminates relation types between given entity pairs, whereas event detection locates event trigger words and recognizes their corresponding event types. For example, in the sentence "*Mary died on Thursday in Memphis*", a relation extraction system needs to recognize a "*Place of Death*" relation between the given entity pair *[Mary, Memphis]*, and an event detection system needs to detect an event trigger "died" and its corresponding event type "*Life:Die*". Relation extraction and event detection both suffer from a data imbalance problem, although existing methods have achieved state-of-the-art performance. For instance, the percentage of non-relation entity pairs is 79.5% in the TACRED dataset [1], and there is no event trigger in over 76% of sentences in the ACE 2005 English dataset. The large negative/positive

* Corresponding author.
E-mail addresses: sdd@bit.edu.cn (D. Song), xujing@bit.edu.cn (J. Xu), pangjinhui@bit.edu.cn (J. Pang), hhy63@bit.edu.cn (H. Huang).

https://doi.org/10.1016/j.ins.2021.05.045
0020-0255/

instance ratio makes it difficult for existing methods to capture the distinct features of positive instances effectively, and therefore these methods perform poorly at distinguishing positive instances from negative ones. To alleviate this problem, emphasizing the characteristics of positive instances or reducing the negative effect of the extreme positive/negative instance ratio seems to be a solution. Without considering non-relation entity pairs, [2] used a pairwise ranking loss to capture the common features of positive instances for relation extraction. To strengthen the influence of event labels, [3] used a bias function for event detection. [4] proposed a multi-task framework by adding an extra relation identification task with a weighted loss function, thus taking advantage of the characteristics of all the positive instances. However, these methods relied on well-designed loss functions and needed extra hyper-parameters to fit different degrees of imbalance for different datasets, which hindered their scalability. Furthermore, these methods could only benefit specific tasks and did not provide unified frameworks across relation extraction and event detection. Therefore, this study has explored a unified framework across relation extraction and event detection to alleviate the data imbalance problem without extra hyper-parameters, thus promoting higher performance in relation extraction or event detection.

Unlike existing methods that focus on adjusting the effect of positive/negative instances, this study explores how the data imbalance problem influences overall performance on extraction tasks and further explores ways to alleviate this influence directly. Empirical results have demonstrated that the data imbalance problem leads to a situation where existing approaches tend to assign negative labels to positive instances and vice versa [4]. In other words, the data imbalance problem limits the ability of existing approaches to identify whether a given instance is positive or negative. This identification problem further limits overall performance (identifying whether each instance is positive or negative and classifying the positive instances into correct types). Hence, improving performance on this identification task can be regarded as a direct solution to the data imbalance problem and should promote better overall performance on extraction tasks.

To improve identification performance on extraction tasks, some concepts are introduced based on sentence A and sentence B, as shown in Fig. 1. A *Certain Type* means that the sentence is not the special type *NONE*, which represents a non-relation or non-event; Identification means that whether one instance is positive or negative is known; Sentence-level identification means that whether any *Certain Types* are present in the sentence is known. As Table 1 shows, when sentence-level

Sentence A
Mary died on Thursday in Memphis.
Event Detection
Certain Types: Death
Identification: "*died*" triggers a event.
Sentence-level identification: Sentence A has triggers which trigger events
Relation Extraction
Certain Types: Place_Death
Identification: [*Mary*, *Memphis*] has a relation.
Sentence-level identification: Sentence A has entity pairs which exist relations.
Sentence B
Mary lives in Memphis.
Event Detection
Certain Types: None
Identification: "*lives*" does not trigger any events.
Sentence-level identification: Sentence B has no trigger which triggers events.
Relation Extraction
Certain Types: Place_Lived
Identification: [*Mary*, *Memphis*] has a relation.
Sentence-level identification: Sentence B has entity pairs which exist relations.

Fig. 1. Example of identification and sentence-level identification.

Table 1
F1-scores of existing approaches on trigger identification (identification performance) and trigger classification (overall performance) using the ACE 2005 English corpus for event detection. TI and TC refer to trigger identification and trigger classification, respectively; TI+ and TC+ refer to trigger identification and trigger classification with sentence-level identification information, respectively; Bi-LSTM refers to bidirectional LSTM; JMEE [5] refers to integrating the GCN with self-attention; MOGANED [6] refers to modeling multi-order syntactic representations using the GCN with aggregative attention; EE-GCN [7] refers the GCN with an edge-aware node update module.

Model	TI	TI+	TC	TC+
Bi-LSTM	70.1	78.6	67.8	71.7
JMEE	75.2	79.4	72.8	75.4
MOGANED	75.9	79.8	73.4	76.6
EE-GCN	78.3	81.8	77.6	79.2

identification information is added for event detection, the identification performance of existing approaches can be significantly enhanced, and this improved identification further enhances overall performance. As for relation extraction, adding sentence-level identification information directly improves identification performance. Therefore, sentence-level identification information can improve identification performance, which further improves overall performance for relation extraction or event detection with imbalanced data. The biggest challenge facing this study was that this sentence-level identification information can be obtained only during training because it is derived from true labels. Intuitively, knowledge distillation methods must be considered because they can transfer knowledge from one model to another. The basic knowledge distillation method uses a teacher network's soft target distribution as the student network's supervised information, thus distilling and transferring knowledge from a teacher network to a student network [8]. However, for relation extraction or event detection, the teacher network's soft target distribution contains little information because it includes too many negative instances. In this paper, a Classifier-Adaptation Knowledge Distillation (CAKD) framework is proposed to effectively enhance performance on relation extraction or event detection by alleviating the data imbalance problem. First, a teacher network is trained, and the sentence-level identification information is clearly given as part of the input. As a result, the sentence-level identification information is possessed by the teacher network. Then the classifier of the teacher network is trained in consistency with the input space. At this point, the classifier, which latently possesses the sentence-level identification information, is frozen. Next, the ultimate neural network is regarded as the student network and is trained without any sentence-level identification information. However, in the process of training the student network, the frozen classifier guides the student network's adaptation and makes the salient semantic information in the student network's extracted features consistent with the information in the teacher network's extracted features. By ensuring consistent salient semantic information, the sentence-level identification information is transferred from the teacher network to the student network, thus improving identification performance and further enhancing the performance of the target neural network. The main contributions of this paper are as follows:

1. To the best of our knowledge, this study is the first to use knowledge distillation to alleviate the imbalanced data problem.
2. A novel knowledge distillation framework (CAKD) is proposed that can not only automatically adapt to different datasets with various positive/negative instance ratios, but also can effectively benefit relation extraction and event detection, which are both challenging and vital information extraction tasks.
3. Experiments were conducted on the TACRED corpus for relation extraction and on the ACE 2005 English corpus for event detection. The results demonstrate the effectiveness of the proposed CAKD framework.
4. Because the proposed CAKD framework can adapt effectively to different extraction tasks including token-level tasks and sentence-level tasks with different baseline models, it can be generalized to other similar extraction tasks.

The rest of this paper is structured as follows: the proposed method is presented in Section 2, experimental results are described in Section 3, extensional experiments with a long-tail distribution are introduced in Section 4, the limitations of the proposed method are presented in Section 5, and related work is discussed in Section 6.

2. Methodology

2.1. Task statement

Given one n-token sentence $\{w_1, w_2, \ldots, w_n\}$ and one entity pair (e_i, e_j), relation extraction predicts the relation type for this entity pair, whereas event detection involves assigning one event label to each token w_i.

The task aims to propose a unified framework to improve relation extraction or event detection performance. The unified framework can fit relation extraction or event detection, but cannot be used to jointly extract relations and detect events.

2.2. Overview

As Fig. 2 shows, the proposed Classifier-Adaptation Knowledge Distillation (CAKD) framework consists of a teacher network and a student network and aims to alleviate the data imbalance problem for relation extraction or event detection. First, the teacher network incorporates sentence-level identification information by adding sentence-level identification embeddings to the input layer during training. Then the teacher network's classifier is frozen, but guides the student network's salient semantic information to be consistent with that of the teacher network, thus helping the student network to capture sentence-level identification information from raw sentences automatically. Thus, the student network enhances its performance on relation extraction or event detection with the help of sentence-level identification information. The red dotted line represents the flow of the sentence-level identification information, which is transferred from the teacher network to the student network. In addition, another classifier is used to predict final extraction results. This framework will be described in detail below.

2.3. The teacher network

The teacher network aims to incorporate sentence-level identification information during training. The sentence-level identification information is directly derived from sentences in the training set and does not rely on any external knowledge or resources. The information is obtained according to whether each sentence contains any Certain Types and is represented by the tag *Positive*/*Negative*. In other words, if no Certain Type is present, the sentence is labeled as *Negative*. Similarly, the sentence is labeled as *Positive* if one or more Certain Types are present. For instance, the following two sentences are included in the training set for event detection: "*Mary died on Thursday in Memphis*" and "*Mary lives in Memphis*". Because these two sentences belong to the training set, it is known that "died" triggers the "*Life:Die*" event in the former sentence, but that no word triggers an event in the latter sentence. Thus, the sentence-level identification is obtained directly: the former sentence contains a trigger that triggers events, but the latter sentence has no trigger that triggers events. Based on this sentence-level identification, the former sentence is labeled as *Positive*, and the latter sentence is labeled as *Negative*. To ensure that the teacher network possesses the sentence-level identification information, this information is transformed into sentence-level identification embeddings, and the sentence-level identification embeddings are included in the input of the teacher network. Because it possesses sentence-level identification information, the teacher network can perform extraction tasks well enough. The teacher network contains an input layer, a feature extraction layer, and a shared classifier layer. In the input layer, given the n-token sentence $\{w_1, w_2, \ldots, w_n\}$, the following two embeddings are concentrated together as the teacher network input vector. The input is denoted by $X_{teacher} = [x_1, x_2, \ldots, x_n]$.

- **Ordinary Input Embeddings** represent the tokens' low-level features, which contain different necessary embeddings for different tasks.
- **Sentence-level Identification Embeddings** are used to provide the sentence-level identification information with random initialization. Two tags are defined: *Positive* and *Negative*. Given a sentence, if no Certain Type exists, *Negative* is used to label every token in the sentence. On the contrary, every token is labeled as *Positive*.Next, the input $X_{teacher}$ is fed into the feature extraction layer. In this layer, various feature extractors can be used to extract features. To provide a clear description, a Bi-LSTM is used to extract features as an example. The input $X_{teacher}$ is fed into the Bi-LSTM to extract the feature representation $F_{teacher}$. Specifically, $F_{teacher}$ contains $[f_1, f_2, \ldots, f_n]$, which represents all tokens' features in the sentence for event detection, whereas $F_{teacher}$ contains f_n, which is the high-level representation of the sentence for relation extraction.

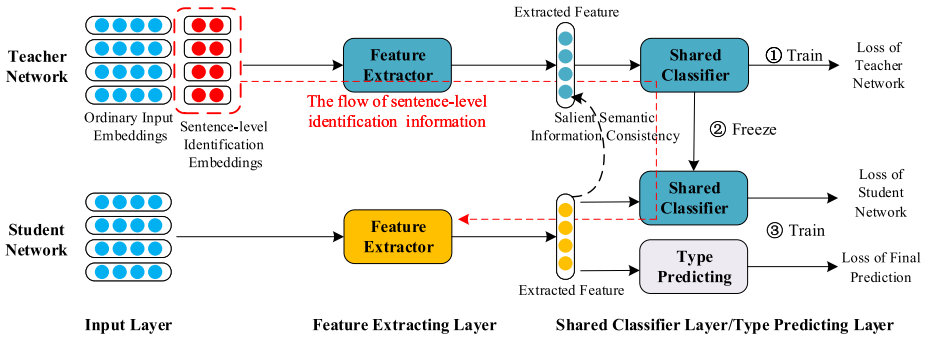

Fig. 2. Overview of the proposed CAKD framework.

$$f_i = \left[\overrightarrow{f_i}, \overleftarrow{f_i}\right], \tag{1}$$

$$\overrightarrow{f_i} = LSTM\left(x_i, \overrightarrow{f_{i-1}}\right), \tag{2}$$

$$\overleftarrow{f_i} = LSTM\left(x_i, \overleftarrow{f_{i+1}}\right). \tag{3}$$

Then the shared classifier layer (i.e., a multi-class classifier C_{share}) predicts the corresponding types given the sentence. The extracted feature representation $F_{teacher}$ is fed into C_{share} to produce a vector $o_{teacher}$, which represents the confidence of candidate types. The prediction probability for the tth type $q_{teacher}(t|\Theta)$ is computed as:

$$o_{teacher} = softmax(W_{share} \cdot F_{teacher} + b_{share}), \tag{4}$$

$$q_{teacher}(t|\Theta) = o_{teacher(t)}, \tag{5}$$

where $o_{teacher(t)}$ represents the tth element of $o_{teacher}$; W_{share} and b_{share} are model parameters of $C_{share}$to be learned; and Θ represents the overall parameters. Hence, the loss of the teacher network is calculated as follows given the training set:

$$loss_{teacher} = -\sum_{k \in K}\sum_{t \in T} p(t)\log(q_{teacher}(t)), \tag{6}$$

where K is the training set; T is the set of all extracted types; and p represents the given instance's true label, which is a one-hot vector. Note that every instance's loss is summed directly, without any hyper-parameters.

2.4. The student network

The student network is the ultimate model that is applied for relation extraction or event detection. Because sentence-level identification information can help existing models to reduce errors caused by the data imbalance problem, but can only be obtained during training, the teacher network is used, which stores the sentence-level identification information during training to guide the student network to learn the sentence-level identification information automatically during the test process. The teacher network transfers the sentence-level identification information to the student network by means of the shared classifier during training, and the student network latently learns to capture the sentence-level identification information automatically from raw text without relying on having this information in the input. With the cooperation of the teacher and student networks, the student network can automatically acquire the sentence-level identification information during the test process and further alleviate the data imbalance problem. This network includes an input layer, a feature extraction layer, a shared classifier layer, and a type prediction layer. The input layer takes raw texts as input without any sentence-level identification information. For the feature extraction layer, various effective feature extractors such as Bi-LSTM and GCNs can be chosen. The shared classifier layer is shared by the teacher network to instruct the student network's feature extraction layer to extract salient semantic information that is consistent with that of the teacher network, thus transferring the sentence-level identification information from the teacher network to the student network. Furthermore, to make each classifier perform its own tasks better, the type prediction layer (i.e., another multi-class classifier) is used to predict the possible types for relation extraction or event detection, enabling the shared classifier layer to focus on transferring the sentence-level identification information. Compared with the teacher network, sentence-level identification embeddings are excluded, and ordinary input embeddings are reserved in the input layer. The input is denoted by $X_{student}$. For the feature extraction layer, different feature extractors that correspond to the teacher network are used. Similarly, another LSTM is used to extract features that are obtained in the same way as in the teacher network. The extracted features are denoted by $F_{student}$in the student network. Then the extracted features $F_{student}$ are directly fed into the classifier C_{share}shared by the teacher network, thus calculating the loss of the student network analogously, which is denoted by $loss_{student}$:

$$o_{student} = softmax(W_{share} \cdot F_{student} + b_{share}), \tag{7}$$

$$q_{student}(t|\Theta) = o_{student(t)}, \tag{8}$$

$$loss_{student} = -\sum_{k \in K}\sum_{t \in T} p(t)\log(q_{student}(t)), \tag{9}$$

where W_{share} and b_{share} are parameters learned by the teacher network and all other parameters are adjusted by the student network. The extracted features $F_{student}$ are fed into another classifier C_{pred}in the type prediction layer, thus obtaining o_{pred}, which indicates the ultimate prediction probabilities of the different types and the corresponding $loss_{pred}$ of each:

$$o_{pred} = softmax(W_{pred} \cdot F_{student} + b_{pred}), \tag{10}$$

$$q_{pred}(t|\Theta) = o_{pred(t)}, \tag{11}$$

$$loss_{pred} = -\sum_{k \in K}\sum_{t \in T} p(t)\log(q_{pred}(t)), \tag{12}$$

where W_{pred} and b_{pred} are the parameters of the classifier C_{pred}.

2.5. Training and test strategy

First, the teacher network is trained by $loss_{teacher}$. Because the teacher network's input layer includes sentence-level identification information, the shared classifier C_{share} can acquire this information. Then the shared classifier C_{share} is frozen, and the student network is trained by $loss_{sum}$:

$$loss_{sum} = loss_{student} + loss_{pred}. \tag{13}$$

On the one hand, $loss_{student}$ is minimized to make $F_{student}$ adapt the classifier C_{share} to learn to extract the sentence-level identification information automatically during training. On the other hand, $F_{student}$ and C_{pred} can become aware of the ground-truth types of instances by minimizing $loss_{pred}$. In the test process, C_{pred} is used to obtain prediction results for relation extraction or event detection.

2.6. Two application tasks

The proposed CAKD framework was applied to two information extraction tasks: relation extraction and event detection. Both relation extraction and event detection are formalized as multi-class classification problems, but they are performed at different levels. Given a sentence, each entity pair is classified as a certain relation type for relation extraction, and each token is classified to a certain event type for event detection. The implementation differences between relation extraction and event detection are described below. In the input layer, ordinary input embeddings include word embeddings, POS embeddings, and NER embeddings for relation extraction. Word embeddings and NER embeddings constitute ordinary input embeddings in the event detection task. In the feature extraction layer, the extracted feature $F_{teacher}/F_{student}$ is a vector containing a high-level representation of the sentence for relation extraction, whereas in the event detection task, the extracted feature $F_{teacher}/F_{student}$ is a matrix that represents all tokens' features in the given sentence. Except for these input and output differences and the corresponding use of different feature extractors for different tasks, the proposed framework is unified and can accommodate relation extraction or event detection.

3. Experiments

This section demonstrates the effectiveness of the proposed CAKD framework in enhancing relation extraction and event detection performance with imbalanced data. The micro-averaged precision (P), recall (R), and F1-score (F_1) are used as evaluation metrics.

3.1. Datasets

For relation extraction, the proposed framework was evaluated on TACRED as released in [1], which consists of 106,264 examples and 41 relation types. Note that the released TACRED training set is imbalanced because it includes 13,012 positive instances and 55,112 negative instances. For event detection, the ACE 2005 English corpus, which contains 599 documents and 33 event types, was used. As in previous work [9], we used 529, 40, and 30 documents as training, development, and test sets, respectively. Note also that the training set of the ACE 2005 English corpus is imbalanced because it includes 3073 positive sentences and 10,966 negative sentences.

3.2. Compare with similar methods

3.2.1. Relation extraction

For relation extraction, the performance of the following models was compared with different feature extractors:

- **Baseline**: Only the student network, which does not freeze its classifier, was used; this case therefore degenerated into a baseline model corresponding with the feature extractor.
- **Baseline + MTL [4]**: A multi-task learning framework was used to learn relation identification and relation classification simultaneously. The multi-task learning framework used a shared network to extract shared features for relation identification and relation classification, and therefore the relation identification enhanced relation classification performance with imbalanced data. This model is introduced in detail below.
- **Baseline + CAKD (Ours)**: The teacher network was first trained, and then the sentence-level identification information was transferred to the student network by sharing the classifier C_{share}. Furthermore, another classifier C_{pred} was used to generate the prediction results.
- **Baseline + Sim-CAKD (Ours)**: The proposed CAKD framework was simplified by using the shared classifier C_{share} to obtain the prediction results.

As for Baseline + MTL, following [4], the input layer, which includes word embeddings, position embeddings, and BIO tag embeddings, was first used to obtain the input embeddings. Given one n-token sentence, the input embedding was denoted

by $X_{mtl} = [x_1, x_2, \ldots, x_n]$. Then X_{mtl} was fed into the selected feature extractor to extract the sentence-level features F_{mtl}. Next, the cross-entropy loss was used to identify relations, whereas the ranking loss was used to classify relations based on the shared features F_{mtl}. To identify relations, the shared features F_{mtl} were fed into a binary classifier $C_{identify}$, which calculated the loss of the relation identification, denoted by $loss_{identify}$:

$$o_{identify} = softmax(W_{identify} \cdot F_{mtl} + b_{identify}), \tag{14}$$

$$q_{identify}(t|\Theta) = o_{identify_{(t)}}, \tag{15}$$

$$loss_{identify} = -\sum(p(1)\log(q_{identify}(1)) + p(0)\log(q_{identify}(0))), \tag{16}$$

where $W_{identify}$ and $b_{identify}$ are model parameters of $C_{identify}$to be learned; p represents the given instance's true identification label, which is a two-dimensional one-hot vector; and Θ represents the overall parameters. For classifying relations, the shared features F_{mtl} were fed into a fully connected layer, which calculated the vector s containing all relations' predicted scores:

$$s = W_{classify} \cdot F_{mtl}. \tag{17}$$

Then the ranking loss $loss_{classify}$ of the relation classification is calculated as follows given the sentence:

$$loss_{+} = \log(1 + exp(\lambda(m^{+} - s(r^{+})))), \tag{18}$$

$$loss_{-} = \log(1 + exp(\lambda(m^{-} + s(r^{-})))), \tag{19}$$

$$loss_{classify} = loss_{+} + loss_{-}, \tag{20}$$

where r^{+} represents the correct relation class; r^{-} represents the chosen incorrect relation class whose score is higher than that of any other relation class except the correct one; m^{+} and m^{-} are margin hyper-parameters; and λ is a scaling value. Based on $loss_{identify}$ and $loss_{classify}$, the multi-task learning framework was trained by the total loss $loss_{multi}$:

$$loss_{multi} = \alpha \cdot loss_{identify} + \beta \cdot loss_{classify}, \tag{21}$$

where α and β represent the weight hyper-parameters.

3.2.2. Event detection

For event detection, various feature extractors were used to compare the performance of Baseline, Baseline + CAKD, and Baseline + Sim-CAKD. Unlike relation extraction, Baseline + MTL, which is a sentence-level approach, could not be applied to event detection directly because event detection is a token-level task. Therefore, inspired by [10], who separated identification from classification to enhance event detection performance, this study used the baseline model not only to classify event types, but also to identify positive and negative types through a multi-task learning framework, denoted by Baseline + MTL (Token). The specific description of Baseline + MTL(Token) refers to Baseline + MTL, with the only difference from Baseline + MTL being that the extracted features are word-level.

3.2.3. Experimental settings

For relation extraction, the following effective feature extractors were chosen to capture sentence representations besides Bi-LSTM, GRU, and Bi-GRU. Note that these feature extractors are specially designed for relation extraction and rely on entity pairs:

- **PA-LSTM**: [1] used a position-aware attention mechanism to explicitly emphasize the corresponding word-based LSTM.
- **GCN**: [11] used a graph convolution network with path-centric pruning to effectively capture the dependency structure of sentences.
- **C-GCN**: [11] combined a graph convolution network with a bi-directional LSTM to capture contextual information.
- **C-AGGCN**: [12] transformed the structure of a dependency tree into a weighted graph to leverage the dependency tree structure and ignore irrelevant information simultaneously.
- **GDPNet**: [24] captured the latent relationships between tokens by using a multi-view graph and concentrates the graph-based and BERT-based representation to predict relations.

Four kinds of embeddings (word embeddings, POS embeddings, NER embeddings, and sentence-level identification embeddings) were used. These four embeddings had dimensions of 300, 30, 30, and 100, respectively. For details of these feature extractors' hyper-parameters, refer to the original papers. The batch size was 50. For event detection, the following effective feature extractors were chosen to capture sentence representations besides Bi-LSTM, GRU, and Bi-GRU. Note that these feature extractors are specially designed for event detection and do not rely on entity pairs:

- **JMEE**: [5] integrated self-attention with a graph convolution network to encode syntactic structures and effectively capture different events' associations.
- **MOGANED**: [6] used a graph convolution network aggregating multi-order syntactic information-based dependency trees to effectively extract dependency information.

- **EE-GCN**: [7] used a graph convolution network with an edge-aware node update module to capture the dependency label information of dependency trees.

Three kinds of embeddings (word embeddings, NER embeddings, and sentence-level identification embeddings) were used. These three embeddings had dimensions of 200, 50, and 300, respectively. Specifically, five kinds of embeddings were used by adding pos-tagging embeddings and positional embeddings for JMEE, and the batch size of MOGANED was set to 8 due to the limitation of the GPUs used, which was more stringent than the original setting. The dimensions of the pos-tagging and positional embeddings were both 50. The hyper-parameters of these three feature extractors followed their original setups. The SGD optimizer was used for both relation extraction and event detection to optimize the networks. The number of teacher network training epochs depends on the teacher network's performance during training. Pytorch was used to implement the proposed framework. Note that the word embeddings were learned by the GloVe or Skip-gram model and the other embeddings were all randomly initialized for both relation extraction and event detection.

3.2.4. Experiment results

For relation extraction, the results shown in Table 2 indicate that Baseline + Sim-CAKD and Baseline + CAKD outperformed their corresponding baseline models using various feature extractors including Bi-LSTM, GRU, Bi-GRU, PA-LSTM, GCN, C-GCN, and GDPNet. This demonstrates that the proposed approach can be regarded as a general framework to improve relation extraction performance. Note that the best F1-score of C-AGGCN was 67.7 when [13] reran the source code and that the proposed C-AGGCN + CAKD model delivered competitive performance compared with C-AGGCN. In addition, Baseline + CAKD performed better than Baseline + Sim-CAKD with most feature extractors, which justifies the assertion that separating type prediction from knowledge transfer leads to greater improvement. As for Baseline + MTL, it enhanced baseline model performance using Bi-LSTM, GRU, and Bi-GRU as feature extractors, although it was not better than Baseline + CAKD. However, it performed worse than the baseline model when using the other feature extractors. Therefore, it can be concluded that Baseline + MTL has a limited ability to generalize this multi-task learning method to different feature extractors. In addition, the proposed framework improves upon baseline models mainly in recall. This is the case because the proposed framework

Table 2
Evaluation of the proposed CAKD framework compared with three similar models on the TACRED dataset for relation extraction. $*$ represents the best F1-score when [13] reruns the published source code.

Model	P	R	F_1
Bi-LSTM	65.7	58.9	62.1
Bi-LSTM + MTL	63.4	62.8	63.1
Bi-LSTM + Sim-CAKD	63.8	62.3	63.0
Bi-LSTM + CAKD	66.2	62.0	**64.1**
GRU	72.2	57.3	63.9
GRU + MTL	65.5	64.0	64.7
GRU + Sim-CAKD	65.8	64.1	64.9
GRU + CAKD	67.7	63.2	**65.4**
Bi-GRU	70.1	59.1	64.2
Bi-GRU + MTL	64.0	65.8	64.9
Bi-GRU + Sim-CAKD	66.7	64.5	65.6
Bi-GRU + CAKD	67.4	64.1	**65.7**
PA-LSTM	65.7	64.5	65.1
PA-LSTM + MTL	66.2	63.4	64.8
PA-LSTM + Sim-CAKD	69.1	61.9	65.3
PA-LSTM + CAKD	67.1	66.4	**66.7**
GCN	69.8	59.0	64.0
GCN + MTL	67.4	59.9	63.4
GCN + Sim-CAKD	68.1	62.5	**65.2**
GCN + CAKD	68.5	61.7	64.9
C-GCN	69.9	63.3	66.4
C-GCN + MTL	69.8	62.2	65.8
C-GCN + Sim-CAKD	70.4	63.5	66.8
C-GCN + CAKD	69.7	65.0	**67.3**
C-AGGCN	71.8	66.4	69.0(67.7$*$)
C-AGGCN + MTL	69.2	64.1	66.6
C-AGGCN + Sim-CAKD	71.6	64.7	68.0
C-AGGCN + CAKD	70.7	65.5	**68.0**
GDPNet	72.0	69.0	70.5
GDPNet + MTL	69.9	66.9	68.4
GDPNet + Sim-CAKD	71.0	70.2	70.6
GDPNet + CAKD	71.3	70.6	**70.9**

learns to extract sentence-level identification information and further helps to identify whether a given instance is positive or negative, which means that the proposed framework tends to be more confident in extracting positive instances. In contrast, the baseline models tend to avoid predicting positive classes without the extracted sentence-level identification information. Therefore, the proposed CAKD framework can improve the recall ratio with different feature extractors. In the event detection task, which is totally different from relation extraction, the experimental results are summarized in Table 3. Overall, these results suggest that the proposed CAKD framework can also enhance baseline model performance by using Bi-LSTM, GRU, Bi-GRU, JMEE, MOGANED, and EE-GCN as feature extractors in the event detection task. Moreover, the results indicate that the proposed CAKD framework outperforms Baseline + MTL(Token) with different feature extractors and that Baseline + MTL(Token) negatively affects baseline model performance using JMEE, MOGANED and EE-GCN as the feature extractor. Furthermore, the improvement in the F1-score can be mainly attributed to the gain in recall. This further justifies the assertion that the teacher network transfers the sentence-level identification information to the student network and that the student network, therefore, becomes more confident in capturing positive instances.

3.3. Compare with SOTA Models

To demonstrate that the proposed CAKD framework delivered competitive performance compared with the state-of-the-art models, it was further compared with the latest relation extraction or event detection models.

3.3.1. Relation extraction

As for relation extraction, the following methods were selected as baselines:

- **LST-AGCN**: [13] transformed dependency tree structure into a weighted graph.
- **Contrastive Pre-training**: [14] explored a contrastive pre-training method to capture relational facts and entity types from context effectively.
- **SpanBERT**: [15] proposed a span-based pretraining method based on BERT.

As shown in Table 4, GDPNet + CAKD outperformed LST-AGCN and Contrastive Pre-training, and delivered comparable performance with SpanBERT. Therefore, the fact is justified that the proposed CAKD framework can also support existing effective feature extractors to achieve competitive performance compared with these state-of-the-art models for relation extraction.

Table 3
Evaluation of the proposed CAKD framework compared with several similar models using the ACE 2005 English corpus for event detection.

Model	P	R	F_1
Bi-LSTM	68.6	67.0	67.8
Bi-LSTM + MTL(Token)	66.4	70.6	68.4
Bi-LSTM + Sim-CAKD	67.2	71.5	69.3
Bi-LSTM + CAKD	68.2	71.7	**69.6**
GRU	67.3	60.6	63.8
GRU + MTL(Token)	67.2	63.2	65.1
GRU + Sim-CAKD	66.6	65.3	**66.0**
GRU + CAKD	65.6	65.6	65.6
Bi-GRU	69.4	68.4	68.9
Bi-GRU + MTL(Token)	67.0	73.3	70.0
Bi-GRU + Sim-CAKD	67.5	73.8	70.5
Bi-GRU + CAKD	68.1	76.2	**71.9**
JMEE	75.3	70.5	72.8
JMEE + MTL(Token)	71.9	65.3	68.5
JMEE + Sim-CAKD	74.0	73.1	73.6
JMEE + CAKD	75.0	72.4	**73.7**
MOGANED	81.0	67.0	73.4
MOGANED + MTL(Token)	76.5	66.1	70.9
MOGANED + Sim-CAKD	79.4	69.1	73.9
MOGANED + CAKD	78.7	72.3	**75.4**
EE-GCN	76.7	78.6	77.6
EE-GCN + MTL(Token)	75.5	75.6	75.5
EE-GCN + Sim-CAKD	76.2	79.1	77.6
EE-GCN + CAKD	76.4	79.9	**78.1**

Table 4
Evaluation of the proposed CAKD framework compared with several SOTA methods using the TACRED dataset for relation extraction.

Model	P	R	F_1
LST-AGCN	–	–	68.8
Contrastive Pre-training	–	–	69.5
SpanBERT	70.8	70.9	70.8
GDPNet + CAKD	71.3	70.6	**70.9**

3.3.2. Event detection

As for event detection, the following methods were selected as baselines:

- **AD-DMBERT**: [25] selected informative training instances to expand event detection datasets by adversarial training.
- **RCEE_ER**: [26] used a machine reading comprehension paradigm to deal with the event detection task.
- **DRMM**: [27] utilized image information for event detection by an alternative dual attention mechanism.

The results shown in Table 5 indicated that EE-GCN + CAKD outperformed AD-DMBERT, RCEE_ER, and DRMM. This further justifies the assertion that the proposed CAKD framework is able to achieve competitive performance compared with these state-of-the-art models when using existing feature extractors.

3.4. Experimental analysis

3.4.1. Effectiveness in distinguishing positive and negative instances

To demonstrate that the proposed CAKD framework can improve the baseline model's ability to identify whether a given instance is positive or negative, which is motivated by the data imbalance problem, the identification performance of the proposed CAKD framework was compared with those of corresponding baseline models. As Table 6 shows, the proposed CAKD framework outperformed the corresponding baseline models using various feature extractors, thus demonstrating that the proposed CAKD framework can improve identification ability. Thanks to this improvement in identification, the data imbalance problem is alleviated.

3.4.2. Effect of various degrees of data imbalance

The proposed CAKD framework's performance was investigated with different degrees of data imbalance by randomly dropping 3000, 6000, 9000, and 12,000 positive instances from the TACRED training set to generate different datasets. Because Bi-LSTM and GRU(Bi-GRU) can be regarded as similar feature extractors, and because Baseline + MTL degrades the performance of baseline models when PA-LSTM, GCN, C-GCN, C-AGGCN, or GDPNet is used, Bi-LSTM was used as the feature extractor to compare the proposed framework with Baseline + MTL with varying degrees of data imbalance. The process of dropping positive instances was repeated three times. For example, 3000 positive instances were randomly dropped three times from TACRED, obtaining three new datasets that had the same number of positive instances in the training set. The performance of each model on T-3000 was calculated by averaging performance on these three datasets. Specifically, precision/recall was calculated by averaging the corresponding precision/recall of these repeated datasets, whereas the F1-score was calculated using the averaged precision and the averaged recall. Table 7 shows the experimental results. In terms of the F1-score, as the positive/negative instance ratios declined, the performance of all variant models dropped. Moreover, the performance gap between Baseline and Baseline + CAKD/Sim-CAKD grew substantially as the data imbalance problem became more serious. This phenomenon indicates that the proposed CAKD framework highlights the effectiveness of alleviating the data imbalance problem compared with the baseline model when the data imbalance problem becomes more serious. Thus, it was indirectly proved that the proposed method can alleviate the data imbalance problem. Moreover, compared with Baseline + MTL, the proposed CAKD framework performed consistently better with different positive/negative instance ratios. This justifies the assertion that the proposed CAKD framework has a better ability to mitigate the data imbalance problem regardless of the degree of imbalance.

Table 5
Evaluation of the proposed CAKD framework compared with several SOTA methods using the ACE 2005 English corpus for event detection.

Model	P	R	F_1
AD-DMBERT	77.9	72.5	75.1
RCEE_ER	75.6	74.2	74.9
DRMM	77.9	74.8	76.3
EE-GCN + CAKD	76.4	79.9	**78.1**

Table 6
Performance of the proposed CAKD framework on the ACE 2005 English corpus for a trigger identification task compared with baseline models.

Model	Baseline	Baseline + CAKD
Bi-LSTM	70.1	71.6
JMEE	75.2	77.2
MOGANED	75.9	78.7
EE-GCN	78.3	79.4

Table 7
Comparison of the proposed CAKD framework with several variant models on datasets with different degrees of data imbalance. Specifically, T-*N* was generated by dropping *N* positive instances from TACRED, and Δ represents the gain of F_1 compared with the Baseline models.

Dataset	Model	P	R	F_1	Δ
TACRED	Baseline	65.7	58.9	62.1	–
	+MTL	63.4	62.8	63.1	1.0
	+Sim-CAKD	63.8	62.3	63.0	0.9
	+CAKD	66.2	62.0	**64.1**	**2.0**
T-3000	Baseline	66.1	55.8	60.5	–
	+MTL	64.0	58.8	61.3	0.8
	+Sim-CAKD	60.1	62.3	61.6	1.1
	+CAKD	62.2	62.3	**62.3**	**1.8**
T-6000	Baseline	68.0	49.0	56.9	–
	+MTL	64.8	52.4	57.9	1.0
	+Sim-CAKD	65.2	52.8	58.4	1.5
	+CAKD	66.6	52.9	**59.0**	**2.1**
T-9000	Baseline	69.9	39.9	50.8	–
	+MTL	62.1	44.6	51.9	1.1
	+Sim-CAKD	68.6	43.9	53.5	2.7
	+CAKD	67.3	45.0	**53.9**	**3.1**
T-12000	Baseline	75.4	17.8	28.8	–
	+MTL	65.2	19.8	30.4	1.6
	+Sim-CAKD	66.5	21.5	**32.5**	**3.7**
	+CAKD	68.8	20.9	32.1	3.3

3.4.3. Case study

In this section, the proposed CAKD framework is shown to improve the ability to identify rare event/relation types or sparse trigger words, thus alleviating the data imbalance problem. As shown in Table 8, two practical experimental results of event detection are presented to compare the proposed CAKD framework with the baseline model. Although "acquitted" obviously triggers the event type "*Acquit*" in S1, the baseline model tends to assign the type "*None*" to "acquitted" due to the rarity of "*Acquit*". In S2, the baseline model also could not recognize "rally" as the trigger of the type "*Demonstrate*" because "rally" is a sparse trigger word. However, the proposed CAKD framework could identify the rare event type "*Acquit*" and the sparse trigger word "rally" because it latently possessed sentence-level identification information. Similarly, Table 9 shows two practical experimental results for relation extraction, where the relation type "*org:parents*" in S1 and "*per:cause_of_death*" in S2 both appear rarely in the training set. The proposed CAKD framework dealt competently with these situations, but the baseline model could not recognize these rare relations effectively. The improved ability to identify rare event/relation types or sparse trigger words also justifies the conclusion that the proposed framework's improvement of the F1-score was mainly attributable to better recall, and therefore the proposed CAKD framework can become more confident in recognizing positive instances.

4. Extension of datasets with long-tail distribution

Because the proposed CAKD framework can alleviate the data imbalance problem effectively, whether the proposed framework can benefit datasets with a long-tail distribution was investigated further. For example, as shown in Fig. 3, many relations suffer from data sparsity, and the numbers of corresponding instances are fewer than 50 in the TACRED dataset. To focus on the long-tail problem, two different settings were used to generate two versions of the variant datasets:

- *Dataset–NONE*: Only non-relation or non-event sentences were dropped in *Dataset*, where *Dataset* means that specific arbitrary dataset.

Table 8
Case study of event detection. The words in bold refer to event triggers.

Sentence	Ground-truth	Baseline	Baseline + CAKD	Frequency
S1: The Pakistani supreme court last year ***acquitted*** *Ayub Masih.*	Acquit	None	Acquit	%0.01
S2: Judge Shahid Rafiq. . .., found Ranjha Masih guilty of defiling Koranic verses during a protest ***rally*** *by the minority Christian community in 1998.*	Demonstrate	None	Demonstrate	%1.47

Table 9
Case study of relation extraction. The bold words refer to entity pairs.

Sentence	Ground-truth	Baseline	Baseline + CAKD	Frequency
S1: The initial offering of AIA raised $ 178 billion for AIG, while the sale of ***ALICO*** to **MetLife** *reaped about $ 155 billion.*	org:parents	None	org:parents	%0.42
S2: Ruben is recovering after ***surgery*** *to his smashed legs, and would be transported to the Netherlands as soon as* ***his*** *medical condition allowed.*	per: cause_of_death	None	per: cause_of_death	%0.17

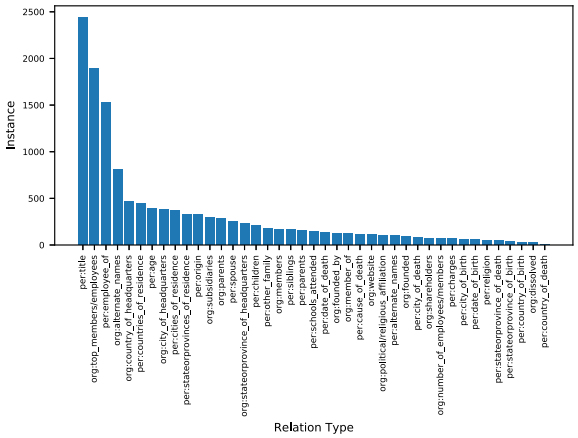

Fig. 3. Distribution of TACRED by relation type.

- *Dataset–K*: Given the specific type *K*, the first step is to pick out all types that have a number of instances greater than or equal to the number of instances of the specific type in the training set of *Dataset* to obtain a type set S_K. Then all sentences containing any types in S_K are dropped to generate *Dataset–K*. For example, given the relation type *Per:employee of*, the set $S_{Per:employeeof}$, which contains *None, Per:title, Org:top members/employees*, was first extracted. Then all sentences containing any types in $S_{Per:employeeof}$ were removed to obtain *Dataset–Per : employeeof*.

For relation extraction, TACRED was chosen as the original dataset. From Fig. 3, *Per:title, Org:top members/employees, Per:employee of, and Org:alternate names*, which all have numerous instances, were picked out to replace *NONE* in training the proposed CAKD framework for *TACRED–NONE*. Note that Bi-LSTM alone was chosen as the feature extractor. As shown in Table 10, the experimental results indicate that the proposed CAKD framework can enhance baseline model performance on datasets with a long-tail distribution, regardless of the relation type chosen. Furthermore, specific types *Per:title, Org:top members/employees, Per:employee of, and Org:alternate names* were chosen to generate the corresponding datasets. Bi-LSTM was chosen as the feature extractor, and only the relation type that had the most instances in the datasets was chosen to replace *NONE*. For example, the relation type *Per:title* was used to replace *NONE* for TACRED–Per:title in the process of training. The results shown in Table 11 indicate that the proposed CAKD also outperformed the baseline model when the degree

Table 10
Performance of the proposed CAKD framework by choosing different relation types to replace the special type *NONE* for *TACRED–NONE*

Relation Types to Replace NONE	Bi-LSTM	Bi-LSTM + CAKD
Per:title	86.7	88.6
Org:top_members/employees	86.7	87.6
Per:employee_of	86.7	87.2
Org:alternate_names	86.7	87.9

Table 11
Performance of the proposed CAKD framework with different variant datasets *TACRED–** generated by choosing different specific relation types

Datasets	Bi-LSTM	Bi-LSTM + CAKD
TACRED–Per:title	86.7	88.6
TACRED–Org:top_members/employees	84.9	85.9
TACRED–Per:employee_of	84.1	84.8
TACRED–Org:alternate names	82.3	82.8

of the long-tail was reduced. These results further justify the assertion that the proposed CAKD framework can deal with datasets having long-tail distributions for relation extraction. For event detection, ACE 2005 English was chosen as the original dataset. Because each sentence that has *Certain Types* also has the type *NONE*, the type *NONE* was reserved, and no other types were chosen to replace it during training. From Fig. 4, specific types *Conflict:Attack, Movement:Transport, Life:Die, Contact:Meet* were chosen to generate corresponding datasets. The results shown in Table 12 demonstrate that the proposed CAKD framework can enhance baseline model performance on datasets with long-tail distributions. Furthermore, it was found that if the event types *Conflict:Attack, Movement:Transport, and Life:Die* were dropped to generate the ACE 2005 English–Contact:Meet dataset, the performance gap between the proposed CAKD framework and the baseline model was greatly reduced. This phenomenon is attributable to the degree of data imbalance. Compared with ACE 2005 English, the ACE 2005 English–Contact:Meet generated dataset became more balanced. Because the proposed framework focuses on the data imbalance problem, it cannot enhance performance much with well-balanced datasets.

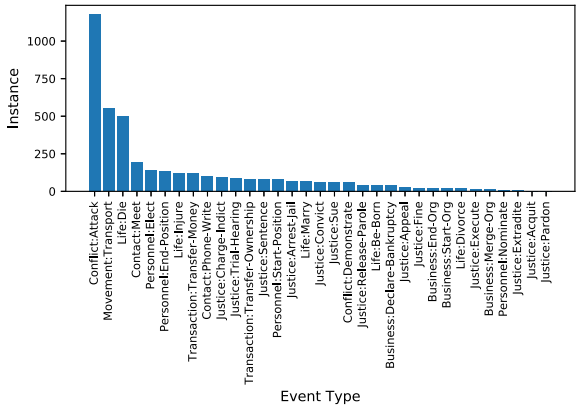

Fig. 4. Distribution of ACE 2005 English by event type.

Table 12
Performance of the proposed CAKD framework with different variant datasets *ACE 2005 English–** generated by choosing different specific event types.

Datasets	Bi-LSTM	Bi-LSTM + CAKD
ACE 2005 English–Conflict:Attack	71.0	72.3
ACE 2005 English–Movement:Transport	57.0	58.0
ACE 2005 English–Life:Die	50.2	51.3
ACE 2005 English–Contact:Meet	47.8	48.0

5. Limitations

Although the method proposed in this study can automatically and effectively alleviate different degrees of data imbalance for relation extraction or event detection, there is limitation on non-extraction tasks. The present study aimed to deal with extraction tasks, but these tasks have too many negative instances. With abundant negative instances, the soft target distribution from the teacher network in conventional knowledge distillation contains little information. The classifier-adaptation knowledge distillation method proposed here can transfer information effectively. However, the proposed classifier-adaptation knowledge distillation is specifically designed for tasks that suffer from the data imbalance problem, but is not a good fit for non-extraction tasks such as text classification, which can use more information from the teacher network's predicted probability distribution due to their more balanced datasets.

6. Related work

6.1. Relation extraction

For relation extraction, traditional kernel-based or feature-based approaches depend on the quality of hand-crafted features and lack of generalization. Therefore, an increasing number of neural-based approaches have been proposed recently. [16] used a convolutional neural network with multiple filter sizes to adapt to imbalanced data. [1] built an LSTM-based model with a position-aware attention mechanism to explicitly emphasize corresponding words. PCA/CNNs and SVMs were integrated to extract relations from massive news texts by [17]. To alleviate the data imbalance problem for distant supervised relation extraction, [18] adjusted the reachable cost of misclassification by applying the silhouette score to measure class-to-class separability. [19] used high-quality negative classes generated by GAN to enhance the performance of distant supervised relation extraction. [20] combined tree-structured LSTM with attention to obtain structure features of the dependency tree and word-based features. Then these features promoted the performance of semantic relation extraction. To take good advantage of the semantic features about the document for document-level relation extraction, [21] used entity pairs to capture the key features among multiple sentences and integrated these features with document-level features by gating mechanism. [22] utilized domain-specific knowledge and multiple source embeddings to generate meta-embeddings in an unsupervised manner, thus benefiting relation extraction. In recent years, GCNs have also been widely used for relation extraction because they can effectively encode information on dependency structures. [11] used a contextualized GCN with path-centric pruning to demonstrate better performance than sequential models. To ignore irrelevant information and leverage the dependency tree structure simultaneously, [13] transformed this structure into a weighted graph. Unlike approaches that rely on structured input, [23] proposed a generalized GCN that can encode unstructured information to obtain the edge parameters of graphs, thus adapting to unstructured input.

6.2. Event detection

As for event detection, feature-based methods rely on human-made lexical features that resemble relation extraction [32], whereas representation-based methods use neural networks to extract semantic information effectively. [9] captured salient event information from sentences using a CNN with a dynamic pooling operation. [28] proposed a unified framework to identify event triggers and corresponding arguments by integrating global and local contexts effectively for biomedical event extraction. [29] utilized dependency tree to capture syntactic features effectively by proposing a tree-based neural network. To reduce noise from unlabeled biomedical data, [30] proposed a novel error detection method to obtain high-quality samples. These samples were then added to expand the training data and enhance biomedical event extraction performance. [31] used gated polar attention mechanism to apply dependency representation learning for biomedical event detection, which alleviates sparsity diffusion and dependency weakness of traditional manual dependency embedding. Recently, delicate hybrid networks have been used for event detection, thus integrating the advantages of different sub-networks. [5] integrated self-attention with GCNs to encode syntactic structures and capture different event associations simultaneously. [10] investigated ways to distill and fuse both discrimination and generalization knowledge to alleviate diversity and ambiguity problems. [6] first used a Bi-LSTM to generate contextualized representations, then added a GCN aggregating multi-order syntactic information-based dependency trees to extract dependency information more effectively. Besides, some efforts to use external resources to enhance event detection performance have proven to be effective. [33] relied on previously trained language models to extract events and generate labeled data for event detection. The labeled data further improved the performance of the event detection framework.

6.3. Knowledge distillation

Knowledge distillation, as proposed by [8], aims to distill knowledge from teacher networks and transfer the resulting knowledge to student networks [34]. Knowledge distillation was originally applied to model compression tasks [35]. More recently, it has been widely and effectively used in various tasks. [36] used knowledge distillation to integrate logic rules into neural networks to enhance the performance of sentiment analysis and named entity recognition. [37] used rich fashion

domain knowledge to enhance performance on the clothing matching task with attentive knowledge distillation. To further incorporate rich knowledge rules for the clothing matching problem, [38] encoded these rules in a probabilistic manner based on knowledge distillation. [39] used a teacher network to learn to recognize actions with full videos and then transferred the progressive knowledge to a student network. The student network could deal better with the early action prediction problem using partial videos. [40] used language branch knowledge distillation and self-knowledge distillation to distill linguistic knowledge, thus enhancing the performance of multilingual unsupervised neural machine translation. To promote the performance of clients in the federated learning framework, [41] regarded model output as knowledge and transferred this knowledge between servers and clients. [42] enhanced the teacher network's ability of mining the knowledge from training set with an inter-class correlation regularization. Thus, this enhanced ability further benefited the performance of student network. As for information extraction tasks, [43] presented a GAN-style knowledge distillation method for the event detection task, which learns to capture knowledge from raw data automatically. [44] distilled monolingual model structure knowledge to a multilingual model to improve the performance of the multilingual model on multilingual named entity recognition. [45] used a marking mechanism and knowledge distillation to integrate open-domain knowledge of triggers from unlabeled data to improve event detection performance. The framework proposed here resembles but is also different from that proposed in [45,46]. These earlier studies focused on distilling knowledge from open-domain resources or soft labels for specific tasks, whereas the present work aimed to explore a unified framework to alleviate the data imbalance problem.

6.4. Relevant extraction tasks

Recently, artificial intelligence technology has been widely used in relevant extraction tasks besides relation extraction and event detection. [47] used the power spectrum analysis algorithm and Contourlet to extract texture features from cloud images, thus helping to automatically detect cloud types. [48] investigated ways to extract, combine, and filter dynamic and static features from the Android system. Then these features were used to enhance the detection of Android malware using the multi-dimensional hybrid feature vector. Lesion extraction was used by [49] to better hide reversible data. The hidden reversible data improved the quality of medical images and protected patients' privacy.

7. Conclusions

In this paper, a novel Classifier-Adaptation Knowledge Distillation (CAKD) framework was proposed to alleviate the data imbalance problem. This framework can not only automatically adapt to datasets with different positive/negative instance ratios, but can also effectively benefit relation extraction and event detection, which are both challenging and vital information extraction tasks. Based on knowledge distillation, the framework can help existing models to reduce errors resulting from the data imbalance problem, thus enhancing relation extraction or event detection performance. Experiments on two standard datasets demonstrated the effectiveness of the proposed framework. In future work, inspired by [50], we will explore a better incorporation location for the sentence-level identification embeddings in the proposed framework, thus further improving the quality of the sentence-level identification information transferred from the teacher network and enhancing the performance of the overall framework.

CRediT authorship contribution statement

Dandan Song: Conceptualization, Methodology, Writing - original draft. **Jing Xu:** Methodology, Software, Writing - original draft. **Jinhui Pang:** Project administration. **Heyan Huang:** Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (Grant No. 2020YFC0833402), the National Natural Science Foundation of China (Grant Nos. 61976021, 61672100, and U1811262), and Guizhou Science and Technology Projects (No[2019]2505).

References

[1] Y. Zhang, V. Zhong, D. Chen, G. Angeli, C.D. Manning, Position-aware attention and supervised data improve slot filling, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 35–45.

[2] C. dos Santos, B. Xiang, B. Zhou, 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), 2015, pp. 626–634.
[3] Y. Chen, H. Yang, K. Liu, J. Zhao, Y. Jia, Collective event detection via a hierarchical and bias tagging networks with gated multi-level attention mechanisms, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 1267–1276.
[4] W. Ye, B. Li, R. Xie, Z. Sheng, L. Chen, S. Zhang, Exploiting entity bio tag embeddings and multi-task learning for relation extraction with imbalanced data, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 1351–1360.
[5] X. Liu, Z. Luo, H.-Y. Huang, Jointly multiple events extraction via attention-based graph information aggregation, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 1247–1256.
[6] H. Yan, X. Jin, X. Meng, J. Guo, X. Cheng, Event detection with multi-order graph convolution and aggregated attention, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 5770–5774.
[7] S. Cui, B. Yu, T. Liu, Z. Zhang, X. Wang, J. Shi, Edge-enhanced graph convolution networks for event detection with syntactic relation, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, 2020, pp. 2329–2339.
[8] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, Stat 1050 (2015) 9.
[9] Y. Chen, L. Xu, K. Liu, D. Zeng, J. Zhao, Event extraction via dynamic multi-pooling 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), 2015, pp. 167–176.
[10] Y. Lu, H. Lin, X. Han, L. Sun, Distilling discrimination and generalization knowledge for event detection via delta-representation learning, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 4366–4376.
[11] Y. Zhang, P. Qi, C.D. Manning, Graph convolution over pruned dependency trees improves relation extraction, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 2205–2215.
[12] Z. Guo, Y. Zhang, W. Lu, Attention guided graph convolutional networks for relation extraction, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 241–251.
[13] K. Sun, R. Zhang, Y. Mao, S. Mensah, X. Liu, Relation extraction with convolutional network over learnable syntax-transport graph, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2020, pp. 8928–8935.
[14] H. Peng, T. Gao, X. Han, Y. Lin, P. Li, Z. Liu, M. Sun, J. Zhou, 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, 2020, pp. 3661–3672.
[15] M. Joshi, D. Chen, Y. Liu, D. Weld, L. Zettlemoyer, O. Levy, SpanBERT: Improving pre-training by representing and predicting spans, in: Transactions of the Association for Computational Linguistics, 2020, pp. 64–77..
[16] T.H. Nguyen, R. Grishman, Relation extraction: Perspective from convolutional neural networks, in: Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, 2015, pp. 39–48.
[17] L. Yin, X. Meng, J. Li, J. Sun, Relation extraction for massive news texts, Computers, Materials and Continua 58 (2019) 275–285.
[18] D. Zeng, Y. Xiao, J. Wang, Y. Dai, A.K. Sangaiah, Distant supervised relation extraction with cost-sensitive loss, CMC-Computers Materials & Continua 60 (3) (2019) 1251–1261.
[19] D. Zeng, Y. Dai, F. Li, R.S. Sherratt, J. Wang, Adversarial learning for distant supervised relation extraction, Computers, Materials & Continua 55 (1) (2018) 121–136.
[20] Z. Geng, G. Chen, Y. Han, G. Lu, F. Li, Semantic relation extraction using sequential and tree-structured LSTM with attention, Information Sciences 509 (2020) 183–192.
[21] C. Yuan, H. Huang, C. Feng, G. Shi, X. Wei, Document-level relation extraction with entity-selection attention, Information Sciences 568 (2021) 163–174.
[22] Q. Liu, J. Lu, G. Zhang, T. Shen, Z. Zhang, H. Huang, Domain-specific meta-embedding with latent semantic structures, Information Sciences 555 (2021) 410–423.
[23] H. Zhu, Y. Lin, Z. Liu, J. Fu, T.-S. Chua, M. Sun, Graph neural networks with generated parameters for relation extraction, in: in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 1331–1339.
[24] F. Xue, A. Sun, H. Zhang, E. Chng, GDPNet: Refining Latent Multi-View Graph for Relation Extraction, arXiv e-prints (2020), pp. arXiv–2012.
[25] X. Wang, X. Han, Z. Liu, M. Sun, P. Li, Adversarial training for weakly supervised event detection, 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), 2019, pp. 998–1008.
[26] J. Liu, Y. Chen, K. Liu, W. Bi, X. Liu, Event extraction as machine reading comprehension, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 1641–1651.
[27] M. Tong, S. Wang, Y. Cao, B. Xu, J. Li, L. Hou, T. Chua, Image enhanced event detection in news articles, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2020, pp. 9040–9047.
[28] W. Zhao, J. Zhang, J. Yang, T. He, H. Ma, Z. Li, A novel joint biomedical event extraction framework via two-level modeling of documents, Information Sciences 550 (2021) 27–40.
[29] H. Fei, Y. Ren, D. Ji, A tree-based neural network model for biomedical event trigger detection, Information Sciences 512 (2020) 175–185.
[30] X. Ma, Y. Lu, Y. Lu, Z. Pei, J. Liu, Biomedical event extraction using a new error detection learning approach based on neural network, CMC-Computers Materials & Continua 63 (2) (2020) 923–941.
[31] L. Li, B. Zhang, Exploiting dependency information to improve biomedical event detection via gated polar attention mechanism, Neurocomputing 421 (2021) 210–221.
[32] M. Sreenivasulu, M. Sridevi, Comparative study of statistical features to detect the target event during disaster, Big Data Mining and Analytics 3 (2020) 121–130.
[33] S. Yang, D. Feng, L. Qiao, Z. Kan, D. Li, Exploring pre-trained language models for event extraction and generation, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 5284–5294.
[34] H. Choi, Y. Lee, K.C. Yow, M. Jeon, Block change learning for knowledge distillation, Information Sciences 513 (2020) 360–371.
[35] C. Buciluǎ, R. Caruana, A. Niculescu-Mizil, Model compression, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006, pp. 535–541.
[36] Z. Hu, X. Ma, Z. Liu, E. Hovy, E. Xing, Harnessing deep neural networks with logic rules, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2016, pp. 2410–2420.
[37] X. Song, F. Feng, X. Han, X. Yang, W. Liu, L. Nie, Neural compatibility modeling with attentive knowledge distillation, in: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, 2018, pp. 5–14.
[38] X. Han, X. Song, Y. Yao, X.-S. Xu, L. Nie, Neural compatibility modeling with probabilistic knowledge distillation, IEEE Transactions on Image Processing 29 (2020) 871–882.
[39] X. Wang, J.-F. Hu, J.-H. Lai, J. Zhang, W.-S. Zheng, Progressive teacher-student learning for early action prediction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3556–3565.
[40] H. Sun, R. Wang, K. Chen, M. Utiyama, E. Sumita, T. Zhao, Knowledge distillation for multilingual unsupervised neural machine translation, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 3525–3535.
[41] L. Hu, H. Yan, L. Li, Z. Pan, X. Liu, Z. Zhang, MHAT: An efficient model-heterogenous aggregation training scheme for federated learning, Information Sciences 560 (2021) 493–503.

[42] C. Tan, J. Liu, X. Zhang, Improving knowledge distillation via an expressive teacher, Knowledge-Based Systems 218 (2021) 106837.
[43] J. Liu, Y. Chen, K. Liu, Exploiting the ground-truth: An adversarial imitation based knowledge distillation approach for event detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 6754–6761..
[44] X. Wang, Y. Jiang, N. Bach, T. Wang, F. Huang, K. Tu, Structure-level knowledge distillation for multilingual sequence labeling, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 3317–3330.
[45] M. Tong, B. Xu, S. Wang, Y. Cao, L. Hou, J. Li, J. Xie, Improving event detection via open-domain trigger knowledge, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 5887–5897.
[46] Z. Zhang, X. Shu, B. Yu, T. Liu, J. Zhao, Q. Li, L. Guo, Distilling knowledge from well-informed soft labels for neural relation extraction, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2020, pp. 9620–9627.
[47] X. Chen, S. Zhao, X. Wang, X. Sun, J. Feng, N. Ye, Texture feature extraction method for ground nephogram based on contourlet and the power spectrum analysis algorithm, Computers Materials & Continua 61 (2019) 861–875.
[48] Y. Li, G. Xu, H. Xian, L. Rao, J. Shi, Novel android malware detection method based on multi-dimensional hybrid features extraction and analysis, Intelligent Automation and Soft Computing 25 (3) (2019) 637–647.
[49] X. Xiao, Y. Yang, R. Li, W. Zhang, A novel reversible data hiding scheme based on lesion extraction and with contrast enhancement for medical images, Computers Materials & Continua 60 (1) (2019) 101–115.
[50] B. Xing, L. Liao, D. Song, J. Wang, F. Zhang, Z. Wang, H. Huang, Earlier attention? Aspect-aware lstm for aspect-based sentiment analysis, Proceedings of the 28th International Joint Conference on Artificial Intelligence, 2019, pp. 5313–5319.