

Pair-Based Joint Encoding with Relational Graph Convolutional Networks for Emotion-Cause Pair Extraction

Junlong Liu, Xichen Shang, Qianli Ma*

School of Computer Science and Engineering,
South China University of Technology, Guangzhou, China
junlongliucs@foxmail.com
qianlima@scut.edu.cn*

Abstract

Emotion-cause pair extraction (ECPE) aims to extract emotion clauses and corresponding cause clauses, which have recently received growing attention. Previous methods sequentially encode features with a specified order. They first encode the emotion and cause features for clause extraction and then combine them for pair extraction. This leads to an imbalance in inter-task feature interaction where features extracted later have no direct contact with the former. To address this issue, we propose a novel **Pair-Based Joint Encoding (PBJE)** network, which generates pairs and clauses features simultaneously in a joint feature encoding manner to model the causal relationship in clauses. PBJE can balance the information flow among emotion clauses, cause clauses and pairs. From a multi-relational perspective, we construct a heterogeneous undirected graph and apply the Relational Graph Convolutional Network (RGCN) to capture the various relationship between clauses and the relationship between pairs and clauses. Experimental results show that PBJE achieves state-of-the-art performance on the Chinese benchmark corpus.¹

1 Introduction

Emotion cause extraction (ECE) is a kind of emotion analysis task which is first proposed by Lee et al. (2010) and has developed for a long time (Gui et al., 2017; Li et al., 2018b, 2019; Hu et al., 2021). ECE extracts the cause for the input document and certain emotion labels. However, emotions in the documents need to be annotated in advance, which requires manual input and takes lots of time (Xia and Ding, 2019). Hence, Xia and Ding (2019) proposes a new task called emotion-cause pair extraction (ECPE). Given a document as the input, ECPE extracts the clauses which express emotions and

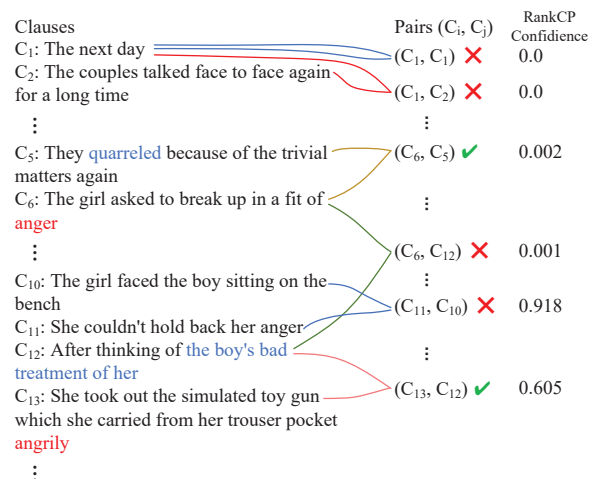


Figure 1: An example document from the ECPE corpus where c_i represents the emotion clause and c_j represents the cause clause in pair. The words in red are the keywords about emotion and the words in blue are about cause. The emotion clause c_6 and c_{12} can not be a pair for lack of causal relationship. We translate it from Chinese into English for ease of reading.

their corresponding clauses which express causes (as shown in Figure 1). Intuitively, ECPE is much more challenging because the clauses classification task and the pairs matching task need to be completed simultaneously.

For ECPE, Xia and Ding (2019) first proposes a two-stage method. However, the two-stage method may cause the problem of error propagation. To solve this problem, the previous work uses end-to-end methods (Ding et al., 2020b; Chen et al., 2020d; Singh et al., 2021). Most of them use sequential encoding, in which their task-specific features are learned sequentially in a predefined order. Specifically, following Wei et al. (2020), ECPE contains two auxiliary tasks, which are emotion clause extraction (EE) and cause clause extraction (CE). The previous work first separately models the clauses for EE and CE. Then they use the clause representations for emotion and cause clauses to model the

*Corresponding author

¹Our codes are publicly available at <https://github.com/tutuDoki/PBJE-ECPE>

pairs for ECPE.

However, the sequential encoding only considers the intra-relationship within pairs or clauses while ignoring the inter-relationship between them. In the sequential encoding, the information can only flow from emotion/cause clause encoder to pair encoder but not vice versa, resulting in the exposure of different amounts of information to pair encoder and clause encoders (Yan et al., 2021; Wang et al., 2022). In this way, if the emotion/cause clause encoder makes incorrect predictions, it will severely misguide the predictions of pair. For example, in the Figure 1, the previous model RankCP (Wei et al., 2020) wrongly extracts the c_{11} as an emotion clause and the c_{10} as a cause clause with high degree of confidence (about 0.98 for c_{11} and 0.94 for c_{10}). Finally, the pair encoder of RankCP extracts the couple (c_{11}, c_{10}) as a candidate pair, which is an incorrect answer. The imbalanced information flow (i.e., from clauses to pairs) might have induced this error.

On the contrary, the joint encoding manner is getting more attention in multi-task learning (Lai et al., 2021; Wang et al., 2020; Fu et al., 2019; Wang and Lu, 2020). It can not only balance the information flow among emotion clauses, cause clauses and pairs to deal with the problems of sequential encoding we mentioned above but also take into account the causal relationship (Chen et al., 2020b) between clauses in ECPE. Since the joint encoding can make the emotion/cause clause encoder and pair encoder interact with each other. During the process of encoding, the clause encoders can pay more attention to whether a clause is suitable for pairs rather than only focus on the information about emotion or cause. The causal relationship is a decisive factor in judging whether emotions and causes match. For example, in Figure 1, c_6 and c_{13} both express anger, and c_{12} is cause clause. However, (c_{13}, c_{12}) is a pair and (c_6, c_{12}) is not. If we separately model the pairs and clauses, the lack of relationship information between these two clauses will increase the difficulty for the model to judge this situation.

Given the above mentioned situation, we propose a novel **Pair-Based Joint Encoding (PBJE)** method, which simultaneously generates pairs and clauses features in a joint feature encoding manner. Specifically, we model the inter-relationship between pairs and clauses, in which a pair only interacts with the corresponding clauses. It helps

pairs learn representations and model the causal relationship from clauses. Meanwhile, the key information about emotion and cause clauses is different. Therefore, different features should be extracted from these two clauses. Considering these multiplex relationships, we construct a heterogeneous undirected graph and apply Relational Graph Convolutional Networks (RGCN) (Schlichtkrull et al., 2018) on it, which includes four kinds of nodes and five kinds of edges, utilizing different approaches to connect the nodes. Thus, it can make the information flow between emotion clauses, between emotion clauses and pairs, etc., more efficient.

We summarize our contributions as follows: (1) We propose a novel method called PBJE to jointly encode the clauses and pairs for ECPE, helping the pairs learn the causal relationship between the two clauses during the encoding process. (2) We propose a RGCN framework to model the multiplex relationship between pairs and clauses. Different edges in the RGCN help the pairs or clauses extract more targeted information, improving the efficiency of the information flow. (3) Experiments on ECPE benchmark corpus demonstrate that PBJE is state-of-the-art.

2 Related Work

2.1 Sequential Encoding

Most of the previous work uses sequential encoding to solve ECPE, including the pipeline and unified framework. Specifically, Xia and Ding (2019) proposes ECPE task and two auxiliary tasks (EE and CE). It uses a two-stage method that first extracts the emotion and cause clauses and then matches them as pairs using Cartesian product for prediction. To address the error propagation problem, Wei et al. (2020) proposes a unified framework that uses Graph Convolution Networks to encode the emotion and cause clauses in the same representations. However, it does not model the pairs, leading to a lack of contextual information in pairs. Furthermore, Ding et al. (2020a,b) and Chen et al. (2020d) build encoders for pairs and clauses separately, which model clauses and then concatenate them as pairs. Considering the symmetric relation between emotion clauses and cause clauses, Cheng et al. (2020) uses a local search method for the clauses which are predicted as emotion clauses or cause clauses.

However, these typical sequential encoding models encode the features in a predefined order, which

leads to the imbalance of the inter-task feature interaction. Since the interaction between clauses and pairs is unidirectional, and the features in pairs can not flow to clauses.

2.2 Implicit Joint Encoding

On the other hand, some work solve ECPE with the implicit joint encoding, such as the sequence labeling methods. We call them "implicit" because these methods joint encode the clauses and pairs, but they do not have the apparent pair features in the model. For example, [Yuan et al. \(2020\)](#) designs a novel cause-pivoted tagging theme. This theme first predicts if a clause is a cause clause, and then finds the corresponding emotion clause among its neighbors using the relative position. In addition, [Fan et al. \(2021\)](#) propose a tag distribution refinement method based on the cause-pivoted sequence labeling, which can leverage the correlations between different tasks (i.e., ECPE, EE, and CE) explicitly and exploit information interaction. Due to the drawback of the cause-pivoted, which can not perfectly extract the cause clauses with multiple emotion clauses, [Chen et al. \(2020c\)](#) designs a more fine-grained tagging scheme that combines emotion tagging and cause tagging with emotion labels separately. However, it still can not handle the situation in which a document has multiple pairs with the same type of emotions. Given this, [Cheng et al. \(2021\)](#) designs a special set of unified labels based on the sequence to sequence model.

Nonetheless, these implicit joint encoding methods based on sequence labeling lack the explicit interaction between clauses and pairs compared with our explicit joint encoding manner. This means that much causal relationship information is ignored in these methods.

3 Task Definition

Given a document $D = (c_1, c_2, \dots, c_N)$ of N clauses and the i -th clauses $c_i = (w_1^i, w_2^i, \dots, w_M^i)$ of M words, ECPE task aims to extract all the emotion-cause pairs in D :

$$P = \{\dots, (c_i, c_j), \dots\} \quad (1 \leq i, j \leq N) \quad (1)$$

where c_i and c_j represent the emotion clause and corresponding cause clause in pairs.

Meanwhile, ECPE has two auxiliary tasks which are emotion clauses extraction (EE) and cause clauses extraction (CE). A clause c_i is emotion

clause if any pair (c_i, c_j) is established. It can be defined as follow:

$$y_i^{emo} = \begin{cases} 1, & \text{if } \exists c_j \in D, (c_i, c_j) \in P \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where $y_i^{emo} = 1$ means c_i is an emotion clause. The extraction of cause clauses is the same as emotion clauses.

4 Proposed Model

In this section, we mainly describe our method, which encodes the pairs and clauses simultaneously and models the causal relationship from clauses in Relational Graph Convolutional Network (RGCN). The structure of PBJE is shown in Figure 2.

4.1 Pair Generator

Following [Wei et al. \(2020\)](#), given a document $D = (c_1, c_2, \dots, c_N)$ consisting of N clauses, we feed D into pre-trained BERT ([Devlin et al., 2019](#)). Specifically, we add a token [CLS] at the beginning and a token [SEP] at the end for each clause and concatenate all clauses together as input. Finally, we use the average pooling of the representations of tokens except for the [CLS] and [SEP] in each clause as the representations of clauses. Hence, the document with N clauses can be represented as:

$$H = \{h_1, h_2, \dots, h_N\} \quad (3)$$

where $h_i \in \mathbb{R}^d$ and d is the hidden size of BERT.

To obtain the representations of pairs, we employ the Pair Generator (PG). Specifically, we concatenate the corresponding two clauses and project them with a learnable relative position embedding:

$$p_{ij} = W_p[h_i, h_j] + b_p + r_{i-j} \quad (4)$$

where $p_{ij} \in \mathbb{R}^d$ represents the pair using c_i as an emotion clause and c_j as a cause clause, $W_p \in \mathbb{R}^{d \times 2d}$ and $b_p \in \mathbb{R}^d$ are learnable parameters, $r_{i-j} \in \mathbb{R}^d$ is the relative position embedding, and $[\]$ denotes the concatenating operation. In addition, following [Wei et al. \(2020\)](#), we set a hyperparameter λ as the local window ($|i - j| \leq \lambda$) to limit the number of pairs.

4.2 Pair-Based Joint Encoder

To balance the interaction between pairs and clauses and capture the causal relationship in pairs, we construct a heterogeneous undirected graph. It can deal with the various relationship between pairs

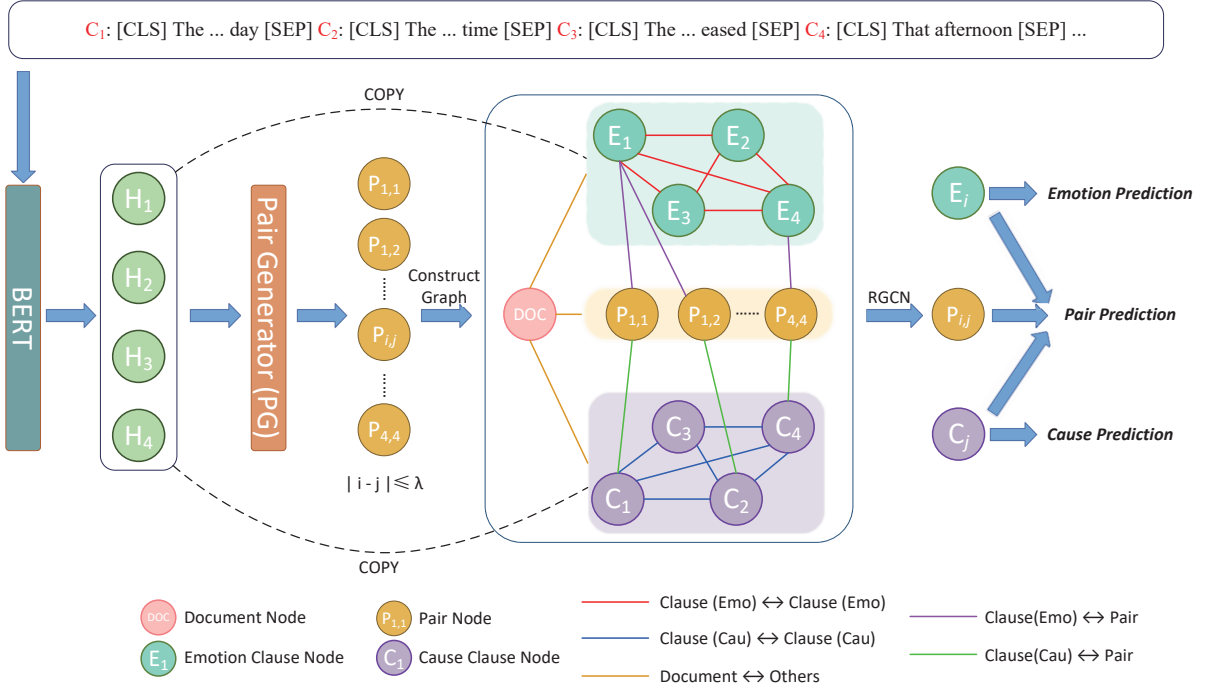


Figure 2: The overall architecture of the PBJE. First, the clauses are input into the pre-trained BERT to get the representations. Then, the representations of the pair are obtained by the pair generator. Next, we construct a heterogeneous undirected graph with the emotion clause nodes, cause clause nodes, pair nodes, and a document node. Finally, after applying RGCN, we use the last layer’s representations of the node for predictions.

and clauses as well as the relationship between clauses efficiently.

The graph has four kinds of nodes: emotion clause nodes, cause clause nodes, pair nodes, and a document node. The emotion information and cause information in a clause are contained in different words. Hence, we separately use two kinds of nodes to represent the emotion clause and the cause clause. In addition, to directly interact with the clauses and capture the causal relationship between the corresponding emotion clause and cause clause, pair nodes are added to the graph. The simultaneous encoding of clauses and pairs balances the information flow between them. Meanwhile, we add a document node to the graph, which can provide some global information (e.g., topics) for the other nodes and interact with others like a pivot.

Moreover, there are mainly five kinds of inter-node edges in our graph:

- **Clause-Clause Edge:** There are two kinds of clause edge in our graph, including the Clause(Emotion)-Clause(Emotion) and Clause(Cause)-Clause(Cause). All emotion/cause clause nodes are fully connected with their own edge. These two edges can help each emotion/cause clause node interact

with other emotion/cause nodes to access contextual information.

- **Clause-Pair Edge:** There are also two kinds of Clause-Pair edge in our graph, including the Clause(Emotion)-Pair and Clause(Cause)-Pair. All pair nodes are connected to their corresponding emotion clause nodes and cause clause nodes with these two kinds of edge. They are the primary way for pairs and clauses to interact with each other and help the emotion and cause nodes to transmit causal relationship to the pair nodes. Besides, the emotion and cause nodes can interact through the pair nodes and these two edges.
- **Document-Others Edge:** The document node is connected to all other nodes with this edge. It can transmit the global information in the document to other nodes and help other nodes ignore the noise from irrelevant nodes.

Furthermore, each type of node has a kind of self-loop edge, which can help each node to maintain its feature in the process of interaction.

With jointly encoding the pairs and clauses at the same level in the graph, the model can solve

the problems in sequential encoding and balance the information between pairs and clauses.

Next, the Relational Graph Convolutional Network (RGCN) (Schlichtkrull et al., 2018) is applied on our heterogeneous undirected graph to aggregate the features from neighbors of each node. First, we use the representations of clause to initialize each emotion and cause clause node:

$$H_E^{(0)} = H, H_C^{(0)} = H \quad (5)$$

where $H_E^{(0)}$ is the representations of emotion clause nodes and $H_C^{(0)}$ is the representations of cause clause nodes. Then, we use the representations of pairs to initialize the pair nodes:

$$H_P^{(0)} = \{p_{11}, p_{12}, \dots, p_{NN}\} \quad (6)$$

In addition, we use the average pooling of all clause representations of document to initialize the document node:

$$H_D^{(0)} = \text{Avgpool}(H) \in \mathbb{R}^d \quad (7)$$

After that, we apply the RGCN on our graph. Given a node u , it is defined as:

$$s_u^{(l)} = W_s^{(l)} h_u^{(l)} + b_s^{(l)} \quad (8)$$

$$t_u^{(l+1)} = s_u^{(l)} + \sum_{r \in \mathcal{R}} \sum_{v \in \mathcal{N}_r(u)} \frac{1}{|\mathcal{N}_r(u)|} W_r^{(l)} h_v^{(l)} + b_r^{(l)} \quad (9)$$

$$h_u^{(l+1)} = \text{ReLU}(t_u^{(l+1)}) \quad (10)$$

where l is the l -th layer of RGCN, \mathcal{R} are different types of edges, $W_s^{(l)} \in \mathbb{R}^{d \times d}$, $b_s^{(l)} \in \mathbb{R}^d$, $W_r^{(l)} \in \mathbb{R}^{d \times d}$ and $b_r^{(l)} \in \mathbb{R}^d$ are learnable parameters, $\mathcal{N}_r(u)$ is the neighbours for node u connected with the edge of type r , and ReLU is the ReLU activation function.

Finally, we select the last layer as the final representation of all nodes after convolutional operation of θ layers:

$$E = H_E^{(\theta)}, C = H_C^{(\theta)}, P = H_P^{(\theta)} \quad (11)$$

4.3 Classification

After getting all the representations of nodes, we use a simple MLP to obtain the predictions of emotion-cause pairs:

$$\hat{y}_{ij}^p = \sigma(\text{MLP}([P_{ij}, E_i, C_j])) \quad (12)$$

Item	Quantity	Percentage(%)
# of documents	1,945	100
- w/ 1 pair	1,746	89.77
- w/ 2 pairs	177	9.10
- w/ ≥ 3 pairs	22	1.13
# of pairs	2167	100
- w/ 0 relative position	511	23.58
- w/ 1 relative position	1342	61.93
- w/ 2 relative position	224	10.34
- w/ ≥ 3 relative position	90	4.15
Avg. # of clauses per document	14.77	
Max. # of clauses per document	73	

Table 1: The detail of the Chinese corpus.

where MLP includes two full-connected layers and a ReLU activation function between them, σ is the sigmoid activation function.

Correspondingly, the binary cross entropy loss is utilized as the loss of ECPE:

$$\mathcal{L}_p = - \sum_i^N \sum_j^N y_{ij}^p \log(\hat{y}_{ij}^p) \quad (13)$$

where y_{ij}^p is the ground truth label.

Following the settings in Wei et al. (2020), we set two auxiliary tasks which are emotion clauses extraction and cause clauses extraction in order to make the clause nodes learn the key contextual information about emotion or cause in the clauses. We compute the probability as follows:

$$\hat{y}_i^e = \sigma(W_e E_i + b_e) \quad (14)$$

$$\hat{y}_j^c = \sigma(W_c C_j + b_c) \quad (15)$$

where \hat{y}_i^e and \hat{y}_j^c are the probability of emotion and cause clauses separately, σ is the sigmoid activation function, $W_e \in \mathbb{R}^{1 \times d}$, $W_c \in \mathbb{R}^{1 \times d}$, $b_e \in \mathbb{R}$ and $b_c \in \mathbb{R}$ are learnable parameters.

Similarly, they have the corresponding loss:

$$\mathcal{L}_e = - \sum_i^N y_i^e \log(\hat{y}_i^e) \quad (16)$$

$$\mathcal{L}_c = - \sum_j^N y_j^c \log(\hat{y}_j^c) \quad (17)$$

where y_i^e and y_j^c are the ground truth labels.

4.4 Training Object

We train PBJE by jointly optimizing the three sub-tasks. The total training object is defined as follow:

$$\mathcal{L} = \alpha \mathcal{L}_p + \beta \mathcal{L}_e + \gamma \mathcal{L}_c \quad (18)$$

where α , β and γ are hyper-parameters.

Approach	Emotion-Cause Pair Extraction			Emotion Clause Extraction			Cause Clause Extraction		
	<i>P</i>	<i>R</i>	<i>F1</i>	<i>P</i>	<i>R</i>	<i>F1</i>	<i>P</i>	<i>R</i>	<i>F1</i>
ECPE-2D	72.92	65.44	68.89	86.27	92.21	<u>89.10</u>	73.36	69.34	71.23
TransECPE	<u>77.08</u>	65.32	70.72	88.79	83.15	85.88	78.74	66.89	72.33
PairGCN	76.92	67.91	72.02	88.57	79.58	83.75	79.07	68.28	73.75
UTOS	73.89	70.62	72.03	88.15	83.21	85.56	76.71	73.20	74.71
MTST-ECPE \diamond	75.78	70.51	72.91	85.83	80.94	83.21	77.64	72.36	74.77
RankCP	71.19	76.30	73.60	91.23	89.99	90.57	74.61	<u>77.88</u>	76.15
ECPE-MLL \dagger	77.00	72.35	<u>74.52</u>	86.08	<u>91.91</u>	88.86	73.82	79.12	<u>76.30</u>
PBJE	79.22	<u>73.84</u>	76.37*	<u>90.77</u>	86.91	88.76	81.79	76.09	78.78

Table 2: The results comparison with baselines on the ECPE corpus for Emotion-Cause Pair Extraction and the two sub-tasks: Emotion clause Extraction and Cause clause Extraction. We introduce these baselines in Appendix A. The best performance is in **bold** and the second best performance is underlined. Approach with \dagger is previous state-of-the-art method. Approach with \diamond is based on our implementation. * denotes $p < 0.0005$ for a two-tailed t-test against the RankCP.

5 Experiments

Extensive experiments are conducted to verify the effectiveness of the PBJE.

5.1 Dataset and Evaluation Metrics

We use the Chinese benchmark dataset released by Xia and Ding (2019), which is constructed from the SINA city news. Table 1 shows the detail of the dataset. Following Xia and Ding (2019), we use the 10-fold cross-validation as the data split strategy and the precision P , recall R and F-score $F1$ as evaluation metrics on three tasks: Emotion-Cause Pair Extraction, Emotion clause Extraction and Cause clause Extraction. We run 10 times and report the average results.

5.2 Implementation Details

We implement PBJE based on Transformers² (Wolf et al., 2020), and use the default parameters in BERT-base-Chinese, setting the hidden size d to 768. Additionally, the hyperparameters λ and θ are set to 3 and 1, respectively. The α , β and γ are all set to 1. We train PBJE through AdamW (Loshchilov and Hutter, 2018) optimizer and the learning rate is $2e-5$. Meanwhile, we add dropout (Srivastava et al., 2014) with a rate of 0.2 to avoid over-fitting. Finally, we set the mini-batch to 4 and the training epoch to 35. The experiments are run on the PyTorch-1.9.0 platform and Ubuntu 18.04 using the Intel(R) Core(TM) i7-8700K CPU, 64GB RAM and NVIDIA GeForce RTX 2080 Ti 11GB GPU.

²<https://github.com/huggingface/transformers>

5.3 Overall Results

Table 2 shows the results of the Emotion-Cause Pair Extraction (ECPE) task and two sub-tasks: Emotion clause Extraction (EE) and Cause clause Extraction (CE). PBJE shows an apparent advantage over previous work, especially on the main task ECPE and auxiliary task CE. We argue that the joint encoding manner plays an important role in PBJE, making the interaction bidirectional and balancing the information obtained by pairs and clauses. It is worth noting that PBJE shows a significant improvement on CE while demonstrating a similar performance on EE compared with ECEP-MLL, which means PBJE can balance the EE and CE. Specifically, RankCP has a huge improvement on EE with applying the sentiment lexicon to PBJE. However, it achieves poor performance on CE, leading to a sharp drop on ECPE. Similarly, ECPE-2D encounters the imbalance problem compared with PBJE. It obtains the second best result on EE, but the worst result on CE. In most cases, EE is more difficult to cope with (Xia and Ding, 2019). Because the expression about cause often contains multiple words, and thus requires the models to understand the text. On the contrary, the expression about emotion only contains a single keyword (e.g., angry, as shown in Figure 1). We argue that the balance benefits from modeling two types of clauses efficiently. And further, this balance helps PBJE improve performance on ECPE.

5.4 Ablation Study

Ablation studies are conducted to verify the effectiveness of the Pair Generator (PG) and different relationship edges and nodes in our graph. Table 3 shows the results of the ablation studies.

Approach	Emotion-Cause Pair Extraction			Emotion Clause Extraction			Cause Clause Extraction		
	<i>P</i>	<i>R</i>	<i>F1</i>	<i>P</i>	<i>R</i>	<i>F1</i>	<i>P</i>	<i>R</i>	<i>F1</i>
PBJE	79.22	73.84	76.37	90.77	86.91	88.76	81.79	76.09	78.78
- w/o Clause-Clause Edge	77.81	<u>73.36</u>	<u>75.45</u>	90.76	<u>87.64</u>	89.14	80.07	75.3	<u>77.54</u>
- w/o Clause-Pair Edge	<u>78.14</u>	72.62	75.21	90.76	86.74	88.66	<u>80.15</u>	74.51	77.16
- w/o Pair Node	76.92	72.37	74.54	89.83	86.62	88.18	79.50	74.81	77.05
- w/o PG	78.02	72.13	74.93	91.22	86.73	<u>88.89</u>	80.07	74.00	76.89
- w/o Pair Node & PG	74.49	73.24	73.76	89.93	87.83	88.82	78.94	<u>75.63</u>	77.18

Table 3: The results of ablation study on the benchmark corpus for emotion-cause pair extraction and the two sub-tasks. The best performance is in **bold** and the second best performance is underlined.

w/o Clause-Clause Edge We use one type of edge to replace the Clause(Emotion)-Clause(Emotion) Edge and the Clause(Cause)-Clause(Cause) Edge. Without these two edges, the performance of our model dramatically drops on CE, further leading to the drop on ECPE. It breaks the balance between EE and CE, meaning the model tends to focus on EE but neglect CE, since EE is the earliest task among these three tasks.

w/o Clause-Pair Edge We remove the Clause(Emotion)-Pair Edge and Clause(Cause)-Pair, and use another edge to replace them. The performance on ECPE is even worse than w/o Clause-Clause Edge. The pairs separately extract the emotions from emotion clauses and the reason from cause clauses. Without this difference, the model can hardly extract information efficiently for the causal relationship.

w/o Pair Node We remove the pair nodes and separately model the emotion and cause clauses using the RGCN. The pairs from PG are utilized to replace the pairs after RGCN. In this way, it is a typical sequential encoding method. Although the PG can still provide some information between emotion and cause clauses, it generates the second worst result on F1. The result shows the importance of joint encoding manner and the causal relationship.

w/o PG Meanwhile, we remove the PG and use another relative position embedding to replace the representations of pairs, which means the pairs with the same relative positions will have the same initial representations in the RGCN and do not contain any clause information. Without the PG, the performance is still better than w/o Pair Node. Despite of absence of clause information, the pairs can learn clause features and causal relationship by the Clause-Pair Edge, which also indicates that the causal relationship is crucial to the modeling of pairs emerging from joint encoding manner.

w/o Pair Node & PG Moreover, we remove the

#Pairs	Approach	<i>P</i>	<i>R</i>	<i>F1</i>
1 per doc.	PBJE	78.44	80.00	79.21
	RankCP	72.03	81.23	76.33
2 or more per doc.	PBJE	83.98	45.29	58.84
	RankCP	67.72	51.46	58.02

Table 4: The results of ECPE for documents with different numbers of pairs.

pair nodes and PG together, similar to the methods in previous work which only encode the clauses for prediction. The *F1* on ECPE sharply dropped by 2.61% and it is the worst model in our experiments. We argue it is caused by the ignorance of pair modeling and the causal relationship in pairs.

In addition, we perform some fine-grained experiments to verify the effect of document node in Appendix B. We believe the information of documents (e.g., topics) is beneficial for the ECPE.

5.5 The Effect of Joint Encoding Manner

To verify the effect of taking into account the joint encoding manner in ECPE, we further conduct some experiments in special cases.

We first compare the results in two situations: documents with one ground truth pair and documents with two or more ground truth pairs. The results are shown in Table 4. PBJE shows clear superiority in both situations. In the documents with a single pair, PBJE demonstrates a significant improvement on *F1* on ECPE, because it avoids the problem caused by sequential encoding. Specifically, in sequential encoding, if the clause encoders extract wrong emotion or cause clauses, the pair encoder is prone to easily group the wrong clauses and extract them as a pair without the bidirectional interaction between clauses and pairs. Apart from that, PBJE also shows a competitive improvement on the document with multiple pairs, which means PBJE can take into account more about the causal relationship and handle the situation mentioned in Figure 1.

Relative Position	Approach	P	R	F
≤ 1	PBJE	80.69	81.26	80.97
	RankCP	77.45	83.38	80.30
≥ 2	PBJE	58.55	28.43	38.28
	RankCP	31.60	32.91	32.24

Table 5: The results of ECPE for pairs of different relative positions.

...It's time for Chinese New Year.(c_4) The creditor removed all his family's grain.(c_5) Other families are celebrating the New Year happily.(c_6) But his family even did not have money for meat. (c_7) His daughter and wife sorrowed. (c_8)...			
PBJE	[c_8, c_7]	RankCP	[c_6, c_5], [c_6, c_7], [c_8, c_7]
Ground Truth	[c_8, c_7]		

Table 6: An example predicted by PBJE and RankCP. The words in **red** are the emotion keywords, and the words in **blue** are the cause keywords. The pairs in **green** are the correct prediction, and the pairs in **red** are incorrect. We translate it from Chinese into English for ease of reading.

In addition, we compare the results in another two situations, which are pairs (c_i, c_j) with relative position less or equal than 1 ($|i - j| \leq 1$) and greater or equal than 2 ($|i - j| \geq 2$). The results are shown in Table 5. Because of the linguistic expression habits, the benchmark dataset has a strong position bias (Ding and Kejriwal, 2020; Xia et al., 2019; Ding et al., 2019). Most of the pairs in the dataset have a relative position within 1 (about 85%). The models can easily achieve a good performance with enough training data in this situation. However, extracting the more difficult pairs are ignored, and these pairs are still prevalent in the real world. Apart from getting a significant improvement in the relative position within 1, PBJE surpasses RankCP in the relative position greater or equal to 2. It shows that PBJE can handle the more complex situation. We argue that the causal relationship in joint encoding manner contributes to PBJE the most. Since PBJE needs to consider the causal relationship among multiple clauses and filter irrelevant clauses, when the relative position is greater than 2 and it can not supply enough information.

5.6 Case Study

We analyze an example selected from the benchmark corpus to demonstrate the effectiveness of joint encoding manner and considering the causal relationship in PBJE, which is shown in Table 6. In addition, we visualize the prediction results in

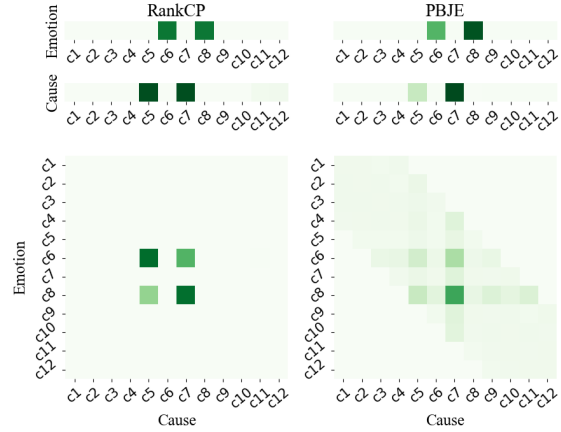


Figure 3: Visualization of the confidence of each prediction in RankCP and PBJE. The deeper color means the higher confidence.

Figure 3.

In this example, RankCP and PBJE both extract the wrong clause c_6 in EE, which might be attributable to the word "happy" in the text. Although c_6 expresses happiness for the Chinese New Year, there is no corresponding cause clause. We do not define it as an emotion clause. Moreover, in CE, RankCP extracts a wrong cause clause c_5 , but PBJE does not (with about 0.25 confidence). The difference shows the clause encoding capability of PBJE. It is worth noting that, in ECPE, the sequential encoding method RankCP tends to easily couple the clause extracted in EE and CE, and it can not correct the errors in EE and CE, resulting in extracting another two wrong pairs. On the contrary, PBJE can avoid them and find out the most likely pair by the joint encoding manner and the balance of information flow between pairs and clauses.

In addition, we analyze more examples in Appendix C. We can find that RankCP tends to extract as many candidate pairs as possible. But there could be many wrong pairs. On the contrary, PBJE tries to extract the correct pairs directly. It can explain why PBJE performs better on precision and worse on recall compared with RankCP.

6 Conclusion

In this paper, we propose a novel **Pair-Based Joint Encoding (PBJE)** network, which encodes the pairs and clauses features simultaneously. It can balance the inter-task feature interaction compared with sequential encoding and model the causal relationship between emotion clauses and corresponding cause clauses by pairs. Furthermore, it can

avoid the wrong predictions in previous tasks in the multi-task learning. From a multi-relational perspective, we propose a Relational Graph Convolutional Network (RGCN) framework to capture the relationship among emotion clauses, cause clauses, pairs, and document, including four types of node and five types of edge. The experiments on the Chinese benchmark corpus show that PBJE achieves state-of-the-art performance.

Limitations

Following the previous work, we implement PBJE setting the hyperparameter λ to 3. It means PBJE can only extract the pairs with a relative distance less than or equal to 3 ($i - j \leq 3$). However, the maximum relative distance of pairs in the dataset is 12. Therefore, no matter how good PBJE is, it can not extract all pairs. There are some methods to solve this problem. For example, we can set λ to 12. Furthermore, we can enumerate all possible pairs without limiting relative distance ($\lambda = +\infty$) to satisfy a larger relative distance that may occur in application. Nonetheless, these two methods will slightly affect the performance of PBJE. Because when we increase λ , there will be more negative samples in the dataset, which exacerbates the problem of data imbalance. On the other hand, because of the language expression, the emotion clauses and cause clauses co-occur most of the time. Setting a large λ is unnecessary. Therefore, the tradeoff between relative distance and performance is what we need to explore in the future. Additionally, a new method which is free from the influence of the relative distance is more desirable.

Acknowledgements

We thank the anonymous reviewers for their helpful feedbacks. The work described in this paper was partially funded by the National Natural Science Foundation of China (Grant Nos. 62272173, 61872148), the Natural Science Foundation of Guangdong Province (Grant Nos. 2022A1515010179, 2019A1515010768).

References

Deli Chen, Yankai Lin, Wei Li, Peng Li, Jie Zhou, and Xu Sun. 2020a. [Measuring and relieving the over-smoothing problem for graph neural networks from the topological view](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):3438–3445.

Xinhong Chen, Qing Li, and Jianping Wang. 2020b. [Conditional causal relationships between emotions and causes in texts](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3111–3121, Online. Association for Computational Linguistics.

Xinhong Chen, Qing Li, and Jianping Wang. 2020c. [A unified sequence labeling model for emotion cause pair extraction](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 208–218, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Ying Chen, Wenjun Hou, Shoushan Li, Caicong Wu, and Xiaoqiang Zhang. 2020d. [End-to-end emotion-cause pair extraction with graph convolutional network](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 198–207, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Zifeng Cheng, Zhiwei Jiang, Yafeng Yin, Na Li, and Qing Gu. 2021. [A unified target-oriented sequence-to-sequence model for emotion-cause pair extraction](#). *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2779–2791.

Zifeng Cheng, Zhiwei Jiang, Yafeng Yin, Hua Yu, and Qing Gu. 2020. [A symmetric local search network for emotion-cause pair extraction](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 139–149, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jiayuan Ding and Mayank Kejriwal. 2020. An experimental study of the effects of position bias on emotion cause extraction. *arXiv preprint arXiv:2007.15066*.

Zixiang Ding, Huihui He, Mengran Zhang, and Rui Xia. 2019. [From independent prediction to reordered prediction: Integrating relative position and global label information to emotion cause identification](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6343–6350.

Zixiang Ding, Rui Xia, and Jianfei Yu. 2020a. [ECPE-2D: Emotion-cause pair extraction based on joint two-dimensional representation, interaction and prediction](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3161–3170, Online. Association for Computational Linguistics.

- Zixiang Ding, Rui Xia, and Jianfei Yu. 2020b. [End-to-end emotion-cause pair extraction based on sliding window multi-label learning](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3574–3583, Online. Association for Computational Linguistics.
- Chuang Fan, Chaofa Yuan, Jiachen Du, Lin Gui, Min Yang, and Ruifeng Xu. 2020. [Transition-based directed graph construction for emotion-cause pair extraction](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3707–3717, Online. Association for Computational Linguistics.
- Chuang Fan, Chaofa Yuan, Lin Gui, Yue Zhang, and Ruifeng Xu. 2021. [Multi-task sequence tagging for emotion-cause pair extraction via tag distribution refinement](#). *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2339–2350.
- Tsu-Jui Fu, Peng-Hsuan Li, and Wei-Yun Ma. 2019. [GraphRel: Modeling text as relational graphs for joint entity and relation extraction](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1409–1418, Florence, Italy. Association for Computational Linguistics.
- Lin Gui, Jiannan Hu, Yulan He, Ruifeng Xu, Qin Lu, and Jiachen Du. 2017. [A question answering approach for emotion cause extraction](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1593–1602, Copenhagen, Denmark. Association for Computational Linguistics.
- Guimin Hu, Guangming Lu, and Yi Zhao. 2021. [Bidirectional hierarchical attention networks based on document-level context for emotion cause extraction](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 558–568, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Tuan Lai, Heng Ji, ChengXiang Zhai, and Quan Hung Tran. 2021. [Joint biomedical entity and relation extraction with knowledge-enhanced collective inference](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6248–6260, Online. Association for Computational Linguistics.
- Sophia Yat Mei Lee, Ying Chen, and Chu-Ren Huang. 2010. [A text-driven rule-based system for emotion cause detection](#). In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 45–53, Los Angeles, CA. Association for Computational Linguistics.
- Qimai Li, Zhichao Han, and Xiao-Ming Wu. 2018a. [Deeper insights into graph convolutional networks for semi-supervised learning](#). In *Thirty-Second AAAI conference on artificial intelligence*.
- Xiangju Li, Shi Feng, Daling Wang, and Yifei Zhang. 2019. [Context-aware emotion cause analysis with multi-attention-based neural network](#). *Knowledge-Based Systems*, 174:205–218.
- Xiangju Li, Kaisong Song, Shi Feng, Daling Wang, and Yifei Zhang. 2018b. [A co-attention neural network model for emotion cause analysis with emotional context awareness](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4752–4757, Brussels, Belgium. Association for Computational Linguistics.
- Ilya Loshchilov and Frank Hutter. 2018. [Decoupled weight decay regularization](#). In *International Conference on Learning Representations*.
- Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. 2020. [Dropedge: Towards deep graph convolutional networks on node classification](#). In *International Conference on Learning Representations*.
- Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. [Modeling relational data with graph convolutional networks](#). In *European semantic web conference*, pages 593–607. Springer.
- Aaditya Singh, Shreeshail Hingane, Saim Wani, and Ashutosh Modi. 2021. [An end-to-end network for emotion-cause pair extraction](#). In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 84–91, Online. Association for Computational Linguistics.
- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. [Dropout: A simple way to prevent neural networks from overfitting](#). *J. Mach. Learn. Res.*, 15(1):1929–1958.
- An Wang, Ao Liu, Hieu Hanh Le, and Haruo Yokota. 2022. [Towards effective multi-task interaction for entity-relation extraction: A unified framework with selection recurrent network](#). *arXiv preprint arXiv:2202.07281*.
- Jue Wang and Wei Lu. 2020. [Two are better than one: Joint entity and relation extraction with table-sequence encoders](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1706–1721, Online. Association for Computational Linguistics.
- Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020. [TPLinker: Single-stage joint extraction of entities and relations through token pair linking](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1572–1582, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Penghui Wei, Jiahao Zhao, and Wenji Mao. 2020. [Effective inter-clause modeling for end-to-end emotion-cause pair extraction](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3171–3181, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Rui Xia and Zixiang Ding. 2019. [Emotion-cause pair extraction: A new task to emotion analysis in texts](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1003–1012, Florence, Italy. Association for Computational Linguistics.

Rui Xia, Mengran Zhang, and Zixiang Ding. 2019. Rthn: A rnn-transformer hierarchical network for emotion cause extraction. In *IJCAI*.

Zhiheng Yan, Chong Zhang, Jinlan Fu, Qi Zhang, and Zhongyu Wei. 2021. [A partition filter network for joint entity and relation extraction](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 185–197, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chaofa Yuan, Chuang Fan, Jianzhu Bao, and Ruifeng Xu. 2020. [Emotion-cause pair extraction as sequence labeling based on a novel tagging scheme](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3568–3573, Online. Association for Computational Linguistics.

Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020. [Graph neural networks: A review of methods and applications](#). *AI Open*, 1:57–81.

Kuangqi Zhou, Yanfei Dong, Kaixin Wang, Wee Sun Lee, Bryan Hooi, Huan Xu, and Jiashi Feng. 2021. [Understanding and Resolving Performance Degradation in Deep Graph Convolutional Networks](#), page 2728–2737. Association for Computing Machinery, New York, NY, USA.

A Comparative Approaches

We compare PBJE with the following methods, which use the pre-trained BERT as encoder:

#Clauses per Doc.	% in Corpus	Approach	<i>P</i>	<i>R</i>	<i>F1</i>
< 14	45.71	PBJE	81.26	75.97	78.53
		- w/o Doc. Node	78.90	72.71	75.68
		RankCP	69.82	77.49	73.46
≥ 14	54.29	PBJE	77.11	72.02	74.48
		- w/o Doc. Node	77.49	70.9	74.05
		RankCP	72.00	75.28	73.60

Table 7: The result of ECPE for documents with different numbers of clauses.

- **ECPE-2D** (Ding et al., 2020a): This method uses the 2D representation to construct a pairs matrix and utilizes the 2D transformer module to interact with other pairs for prediction.
- **TransECPE** (Fan et al., 2020): It is a transition-based method which transforms the task into a procedure of parsing-like directed graph construction.
- **RankCP** (Wei et al., 2020): This method tackles emotion-cause pair extraction from a ranking perspective, which ranks pairs in a document and proposes a one-step neural approach to extract.
- **PairGCN** (Chen et al., 2020d): This method constructs a graph using the pair nodes and a Pair Graph Convolutional Network to model the dependency relations among candidate pairs.
- **ECPE-MLL** (Ding et al., 2020b): It is the current state-of-the-art method, which employs two joint frameworks, including the emotion-pivot cause extraction and cause-pivoted emotion extraction with sliding window strategy.
- **UTOS** (Cheng et al., 2021): It solves this task using sequence labeling, which allows to extract pairs through one pass and addresses the error propagation problem.
- **MTST-ECPE** (Fan et al., 2021): This method uses a multi-task sequence tagging framework with refining the tag distribution.

B The Effect of Document Node

To verify the effect of document node, some extensive experiments are conducted in different lengths of a document, according to the average number of clauses per document 14.77 and the median 14.

As shown in Table 7, the document node can help PBJE to improve the performance on ECPE

ID	Examples	Predicted Pairs		Ground Truths
		PBJE	RankCP	
1	Mr. Zhang, the boss of the repair shop where he used to be, said.(c_1) Mr. Wang was very surprised when he came to apply .(c_2) He was the most educated worker in the garage at the time.(c_3) Every time he fixed the cars, he put his heart into it.(c_4) Once he was ill , he still insisted on repairing the car .(c_5) It made Mr. Zhang so moved .(c_6) ...	$[c_2, c_2]$ $[c_6, c_5]$	$[c_2, c_2]$ $[c_2, c_5]$ $[c_6, c_5]$	$[c_2, c_2]$ $[c_6, c_5]$
		$[c_4, c_4]$ $[c_6, c_6]$	$[c_4, c_4]$ $[c_4, c_6]$ $[c_6, c_6]$	$[c_4, c_4]$ $[c_6, c_6]$
2	... She and her family are very healthy(c_2) So they can continue to donate blood to contribute to the society(c_3) She used to worry about limiting the age of blood donation to 55 years old (c_4) She is used to donating blood now(c_5) If she can't continue donating blood because of her age , she will be very disappointed (c_6) ...	$[c_4, c_4]$ $[c_6, c_6]$	$[c_4, c_4]$ $[c_4, c_6]$ $[c_6, c_6]$	$[c_4, c_4]$ $[c_6, c_6]$

Table 8: Examples predicted by PBJE and RankCP. The words in **red** are the emotion keywords, and the words in **blue** are the cause keywords. The pairs in **green** are the correct prediction, and the pairs in **red** are incorrect. We translate them from Chinese into English for ease of reading.

in both short (< 14) and long (≥ 14) documents. Since the emotion clauses and cause clauses make up a small proportion of the total clauses in each document, even in the short documents. Most of the documents only have 1 pair. Therefore, the fully connected graphs of emotion and cause clauses contain lots of noise. It makes each emotion and cause clause node can hardly learn the effective and enough contextual information. In this situation, the document node can filter the invalid information and integrate them into global information, then transmits them to other nodes through the Document-Others Edge. However, the improvement in short documents on ECPE is much more than in long documents with the help of document node. Because when the document is long, there are too many features of clauses for average pooling. This results in a lower effective information density in long documents than in short documents. Further, it makes the representations of document uncharacteristic and contain noise. But the performance does not drop in this situation since the Document-Others Edge can selectively transmit information through learning ability.

C Additional Case Study

To further demonstrate the importance of considering the causal relationship in RGCN, we analyze another two examples selected from the benchmark corpus. We show them in Table 8

For the first example, although RankCP extracts all the correct pairs, it extracts another incorrect pair (c_2, c_5). The clause c_2 expresses surprise, and the clause c_5 expresses the persistence of fixing cars even when he is sick. Although they are emotion clause and cause clause separately, c_2 is not the reason to cause c_5 obviously. By considering the

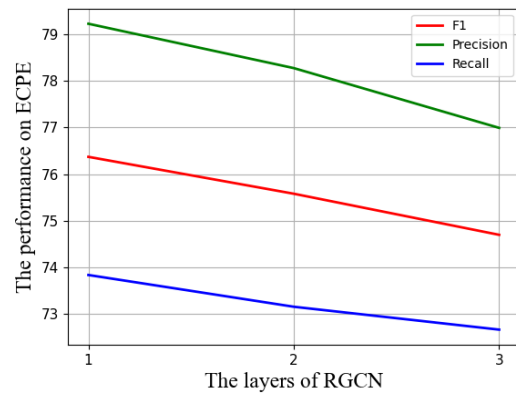


Figure 4: The influence of different θ on ECPE.

causal relationship, PBJE avoids this situation.

Next, for the second example, RankCP encountered the same problem as the first example. Further, the emotion clause c_4 expresses worry, and c_6 expresses disappointment, which are both negative emotions. Moreover, the cause clause c_4 describes the same thing with cause clause c_6 about the age restrictions on blood donation. Therefore, it is more difficult for models to judge in this situation. Nevertheless, PBJE successfully deals with this situation.

D Hyperparameters Discussion

As shown in Figure 4, we examine the effects of different values of θ on ECPE. We can observe that the performance tends to drop with the increase of the layers of RGCN. We argue that the over-smoothing causes this problem (Li et al., 2018a; Zhou et al., 2020; Chen et al., 2020a). Specifically, when the θ is greater than 1, it means that the Relationship Graph Convolutional Network(RGCN) are repeatedly applied. It may mix the features of nodes from

different classes and make them indistinguishable, leading to the drop on ECPE. In addition, more layers indicates more learnable parameters, which will result in over-fitting (Zhou et al., 2021; Rong et al., 2020).