ECPE-2D: Emotion-Cause Pair Extraction based on Joint Two-Dimensional Representation, Interaction and Prediction

Zixiang Ding, Rui Xia*, Jianfei Yu
School of Computer Science and Engineering,
Nanjing University of Science and Technology, China
{dingzixiang,rxia,jfyu}@njust.edu.cn

Abstract

In recent years, a new interesting task, called emotion-cause pair extraction (ECPE), has emerged in the area of text emotion analysis. It aims at extracting the potential pairs of emotions and their corresponding causes in a document. To solve this task, the existing research employed a two-step framework, which first extracts individual emotion set and cause set, and then pair the corresponding emotions and causes. However, such a pipeline of two steps contains some inherent flaws: 1) the modeling does not aim at extracting the final emotion-cause pair directly; 2) the errors from the first step will affect the performance of the second step. To address these shortcomings, in this paper we propose a new end-to-end approach, called ECPE-Two-Dimensional (ECPE-2D), to represent the emotion-cause pairs by a 2D representation scheme. A 2D transformer module and two variants, window-constrained and cross-road 2D transformers, are further proposed to model the interactions of different emotion-cause pairs. The 2D representation, interaction, and prediction are integrated into a joint framework. In addition to the advantages of joint modeling, the experimental results on the benchmark emotion cause corpus show that our approach improves the F1 score of the state-of-the-art from 61.28% to 68.89%.

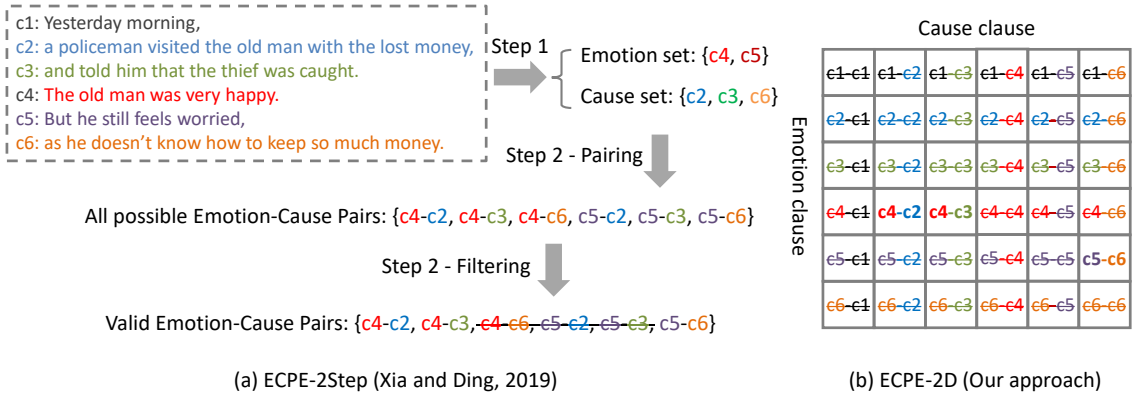
1 Introduction

Emotion cause extraction (ECE), as a sub-task of emotion analysis, aims at extracting the potential causes of certain emotion expressions in text. The ECE task was first proposed by Lee et al. (2010) and defined as a word-level sequence labeling problem. Gui et al. (2016a) released a new corpus and re-formalized the ECE task as a clause-level extraction problem. Given an emotion annotation, the goal of ECE is to predict for each clause in a document if the clause is an emotion cause. This framework has received much attention in the following studies in this direction. Although the ECE task was well defined, it has two problems: Firstly, the emotion must be annotated manually before cause extraction, which greatly limits its practical application; Secondly, the way to first annotate the emotion and then extract the causes ignores the fact that emotions and causes are mutually indicative. To address this problem, we have proposed a new task named emotion-cause pair extraction (ECPE), aiming to extract the potential pairs of emotions and their corresponding causes together in our previous work (Xia and Ding, 2019).

Specifically, ECPE is defined as a fine-grained emotion analysis task, where the goal is to extract a set of valid emotion-cause pairs, given a document consisting of multiple clauses as the input. Figure 1 (a) shows an example of the ECPE task. The input in this example is a document consisting of six clauses. Clause c4 contains a "happy" emotion and it has two corresponding causes: clause c2 ("a policeman visited the old man with the lost money"), and clause c3 ("told him that the thief was caught"). Clause c5 contains a "worried" emotion and the corresponding cause is clause c6 ("as he doesn't know how to keep so much money"). The final output is a set of valid emotion-cause pairs defined at clause level: {c4-c2, c4-c3, c5-c6}. We have also proposed a two-step approach (ECPE-2Steps) to address the ECPE task (Xia and Ding, 2019). ECPE-2Steps is a pipeline of two steps: Step 1 extracts an emotion set and a cause set individually. For example in Figure 1 (a), the emotion set is {c4, c5} and the cause set is {c2, c3, c6}; Step 2 conducts emotion-cause pairing and filtering based on the outputs of Step 1. As shown in Figure 1 (a), it first gets the candidate emotion-cause pairs by applying a Cartesian product to the emotion set and

*Corresponding author

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 3161–3170
July 5 - 10, 2020. ©2020 Association for Computational Linguistics



(a) ECPE-2Step (Xia and Ding, 2019) (b) ECPE-2D (Our approach)

Figure 1: An example showing two frameworks for solving the emotion-cause pair extraction (ECPE) task.

cause set, and then train an independent filter to remove the invalid pairs.

Although the ECPE-2Steps approach seems reasonable and performs well, it still has the following shortcomings: (1) as a pipeline of two separate steps, ECPE-2Steps requires two prediction steps to get the final emotion-cause pair. The training of the model is also not directly aimed at extracting the final emotion-cause pair. (2) The errors from Step 1 will affect the performance of Step 2. For one thing, the upper bound of the recall in Step 2 is determined by the recall in Step 1, because Step 2 cannot produce emotion-cause pairs from the emotions or causes that were not extracted by Step 1; for another, if Step 1 predicts too many incorrect emotions or causes, the precision of Step 2 will be reduced.

To address these problems, in this work we propose a new end-to-end ECPE solution, called ECPE-Two-Dimensional (ECPE-2D), to represent the emotion-cause pairs by a 2D representation scheme, and integrate the **emotion-cause pair representation**, **interaction** and **prediction** into a joint framework. As shown in Figure 1 (b), firstly, we design a 2D representation scheme to represent the emotion-cause pairs in forms of a square matrix, where each item represents an emotion-cause pair. Secondly, a 2D Transformer framework and its two variants, window-constrained and cross-road 2D transformers, are further proposed to capture the interaction between different emotion-cause pairs. Finally, we extract the valid emotion-cause pairs based on the 2D representation by conducting a binary classification on each emotion-cause pair. These three parts are integrated into a unified framework and trained simultaneously.

We evaluate our ECPE-2D approach on the benchmark emotion cause corpus. The experimental results prove that ECPE-2D can obtain overwhelmingly better results than the state-of-the-art methods on the emotion-cause pair extraction task and two auxiliary tasks (emotion extraction and cause extraction).

2 Approach

2.1 Overall Architecture

Following our prior work (Xia and Ding, 2019), we formalize the emotion-cause pair extraction (ECPE) task as follows. The input is a document consisting of multiple clauses $d = [c_1, c_2, \cdots, c_{|d|}]$, the goal of ECPE is to extract a set of emotion-cause pairs in d:

$$P = \{\cdots, c^{\text{emo}}\text{-}c^{\text{cau}}, \cdots\}, \quad (1)$$

where c^{emo} is an emotion clause and c^{cau} is the corresponding cause clause.

The overall architecture of the proposed method is shown in Figure 2. It consists of three parts: 1) 2D Emotion-Cause Pair Representation; 2) 2D Emotion-Cause Pair Interaction; 3) 2D Emotion-Cause Pair Prediction. Firstly, an individual emotion/cause encoding component is firstly employed to obtain the emotion-specific representation vectors and cause-specific representation vectors. A full pairing component is applied to pair the two representation vectors into a 2D representation matrix. Then a 2D transformer module is proposed to model the interactions between different emotion-cause pairs. For each emotion-cause pair in the matrix, the updated representation is finally fed to a softmax layer to predict if the pair is valid or not. The three modules are integrated into a unified framework and trained simultaneously.

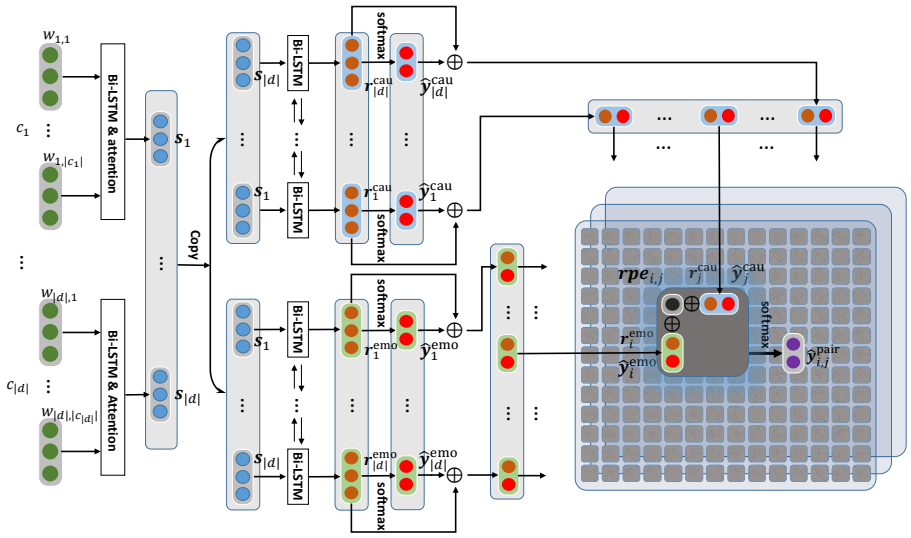


Figure 2: Overview of the proposed joint framework for emotion-cause pair extraction.

2.2 2D Emotion-Cause Pair Representation

2.2.1 Individual Emotion/Cause Encoding

The purpose of the clause encoder layer is to generate an emotion-specific representation and a cause-specific representation for each clause in a document. The input is a document contains multiple clauses: $d = [c_1, c_2, \cdots, c_{|d|}]$, and each clause also contains multiple words $c_i = [w_{i,1}, w_{i,2}, ..., w_{i,|c_i|}]$. A hierarchical neural network which contains two layers is employed to capture such a word-clause-document structure.

The lower layer consists of a set of word-level Bi-LSTM modules, each of which corresponds to one clause and accumulate the context information for each word of the clause. The hidden state of the j-th word in the i-th clause $\mathbf{h}_{i,j}$ is obtained based on a bi-directional LSTM. An attention mechanism is then adopted to get the clause representation $\mathbf{s}_i$.

The upper layer is composed of two independent components, with the goal to generate an emotion-specific representation $\mathbf{r}_i^{\text{emo}}$ and a cause-specific representation $\mathbf{r}_i^{\text{cau}}$ for each clause, respectively. Both components take the clause representation $(\mathbf{s}_1, \mathbf{s}_2, , \mathbf{s}_{|d|})$ as input and use two clause-level Bi-LSTMs to obtain $\mathbf{r}_i^{\text{emo}}$ and $\mathbf{r}_i^{\text{cau}}$, respectively. Finally, $\mathbf{r}_i^{\text{emo}}$ and $\mathbf{r}_i^{\text{cau}}$ are respectively feed into two softmax layers to get the emotion prediction $\hat{\mathbf{y}}_i^{\text{emo}}$ and cause prediction $\hat{\mathbf{y}}_i^{\text{cau}}$:

$$\hat{\mathbf{y}}_i^{\text{emo}} = \text{softmax}(\mathbf{W}^{\text{emo}}\mathbf{r}_i^{\text{emo}} + \mathbf{b}^{\text{emo}}), \quad (2)$$

$$\hat{\mathbf{y}}_i^{\text{cau}} = \text{softmax}(\mathbf{W}^{\text{cau}}\mathbf{r}_i^{\text{cau}} + \mathbf{b}^{\text{cau}}). \quad (3)$$

It should be noted that the individual emotion/cause encoder here is a compatible module. Other emotion/cause encoder such as Inter-CE, Inter-EC (Xia and Ding, 2019), and BERT (Devlin et al., 2019) can also be used. We will compare and discuss them in the experiments.

2.2.2 Emotion-Cause Full Pairing

In contrast to the ECPE-2Steps approach (Xia and Ding, 2019) which only extract pairs from the individual emotion set and cause set, we consider all possible pairs of clauses in d as candidates. Assuming the length of the document is $|d|$, then all possible pairs form a matrix $\boldsymbol{M}$ of the shape $|d| * |d|$, where the rows and columns represent the index of the emotion clause and the cause clause in the document, respectively. c_i^{emo}-c_j^{cau} is the element in the i-th row and the j-th column of $\boldsymbol{M}$ and indicates the emotion-cause pair that consists of the i-th clause and the j-th clause, encoded as:

$$\boldsymbol{M}_{i,j} = \mathbf{r}_i^{\text{emo}} \oplus \hat{\mathbf{y}}_i^{\text{emo}} \oplus \mathbf{r}_j^{\text{cau}} \oplus \hat{\mathbf{y}}_j^{\text{cau}} \oplus \boldsymbol{rpe}_{i,j}, \quad (4)$$

where $\mathbf{r}_i^{\text{emo}}$ and $\hat{\mathbf{y}}_i^{\text{emo}}$ are emotion-specific representation and emotion prediction of the i-th clause c_i, $\mathbf{r}_j^{\text{cau}}$ and $\hat{\mathbf{y}}_j^{\text{cau}}$ are cause-specific representation and cause prediction of the j-th clause c_j. $\boldsymbol{rpe}_{i,j}$ is a relative position embedding vector of c_j relative to c_i.

2.3 2D Emotion-Cause Pair Interaction

In the previous section, we have obtained a 2D representation matrix consisting of all possible emotion-cause pairs. Each element of the matrix represents a specific emotion-cause pair.

Considering that a document of length $|d|$ will generate $|d| * |d|$ possible emotion-cause pairs, a-

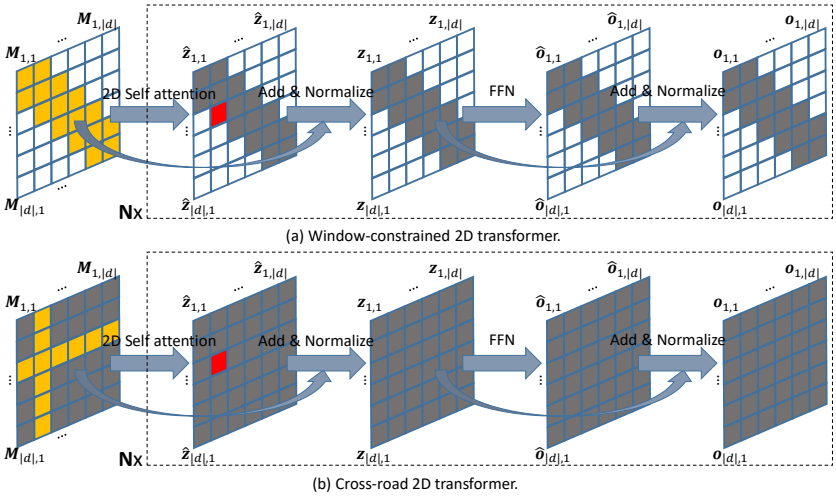


Figure 3: Two simplified versions of 2D transformer for emotion-cause pair interaction.

mong which only a very small number of pairs are positive samples. Using the independent pair representation for emotion-cause pair prediction will not take advantage of this global information. Therefore, we further designed a 2D transformer for the ECPE task to effectively achieve the interaction between emotion-cause pairs.

2.3.1 Standard 2D Transformer

The standard 2D transformer (Vaswani et al., 2017) consists of a stack of N layers. Each layer consists of two sublayers: a multi-head 2D self-attention mechanism followed by a position-wise feed forward network.

Multi-head 2D Self-attention. The multi-head 2D self-attention mechanism first calculates the query vector $\boldsymbol{q}_{i,j}$, key vector $\boldsymbol{k}_{i,j}$ and value vector $\boldsymbol{v}_{i,j}$ for each pair c_i^{emo}-c_j^{cau} in the document d as :

$$\boldsymbol{q}_{i,j} = \text{Relu}(\boldsymbol{M}_{i,j}\boldsymbol{W}_Q), \quad (5)$$

$$\boldsymbol{k}_{i,j} = \text{Relu}(\boldsymbol{M}_{i,j}\boldsymbol{W}_K), \quad (6)$$

$$\boldsymbol{v}_{i,j} = \text{Relu}(\boldsymbol{M}_{i,j}\boldsymbol{W}_V), \quad (7)$$

where $\boldsymbol{W}_Q \in R^{n\times n}$, $\boldsymbol{W}_K \in R^{n\times n}$, $\boldsymbol{W}_V \in R^{n\times n}$ are parameters for queries, keys and values respectively.

For each pair c_i^{emo}-c_j^{cau}, a set of weights $\beta_{i,j} = \{\beta_{i,j,1,1}, \beta_{i,j,1,2}, \cdots, \beta_{i,j,|d|,|d|}\}$ are learned:

$$\beta_{i,j,a,b} = \frac{\exp(\frac{\boldsymbol{q}_{i,j}\cdot\boldsymbol{k}_{a,b}}{\sqrt{n}})}{\sum_{a'}\sum_{b'}\exp(\frac{\boldsymbol{q}_{i,j}\cdot\boldsymbol{k}_{a',b'}}{\sqrt{n}})}. \quad (8)$$

Then the new feature representation of c_i^{emo}-c_j^{cau} is obtained by considering all the $|d| * |d|$ pairs in M:

$$\hat{\boldsymbol{z}}_{i,j} = \sum_{a=1}^{|d|}\sum_{b=1}^{|d|}\beta_{i,j,a,b}\cdot\boldsymbol{v}_{a,b}. \quad (9)$$

Position-wise Feed Forward Network. In addition to the attention sublayer, a position-wise feed forward network is applied to each pair separately and identically:

$$\hat{\mathbf{o}}_{i,j} = \max(0, \boldsymbol{z}_{i,j}\boldsymbol{W}_1 + \boldsymbol{b}_1)\boldsymbol{W}_2 + \boldsymbol{b}_2. \quad (10)$$

It should be noted that both of the above two sublayers use the residual connection followed by normalization layer at its output:

$$\boldsymbol{z}_{i,j} = \text{Normalize}(\hat{\boldsymbol{z}}_{i,j} + \boldsymbol{M}_{i,j}), \quad (11)$$

$$\boldsymbol{o}_{i,j} = \text{Normalize}(\hat{\mathbf{o}}_{i,j} + \boldsymbol{z}_{i,j}). \quad (12)$$

As has mentioned, the standard 2D transformer consists of a stack of N layers. Let l denotes the index of transformer layers. The output of the previous layer will be used as the input of the next layer:

$$\boldsymbol{M}_{i,j}^{(l+1)} = \boldsymbol{o}_{i,j}^{(l)}. \quad (13)$$

Computational inefficiency. Since the outputs of the standard transformer are $|d| * |d|$ elements, each element requires the calculation of $|d| * |d|$ attention weights, and eventually $(|d|*|d|)*(|d|*|d|)$ weights are needed to be calculated and temporarily stored. To alleviate the computational load, we furthermore propose two variants of the standard 2D Transformer in the following two subsections: 1) window-constrained 2D Transformer and 2) cross-road 2D Transformer, as shown in Figure 3.

2D transformer	Time complexity	Space complexity
Standard	$O(batch * \|d\| * \|d\| * n * (\|d\| * \|d\| + n))$	$O(batch * \|d\| * \|d\| * (\|d\| * \|d\| + n))$
Window-constrained	$O(batch * \|d\| * w * n * (\|d\| * w + n))$	$O(batch * \|d\| * w * (\|d\| * w + n))$
Cross-road	$O(batch * \|d\| * \|d\| * n * (\|d\| + n))$	$O(batch * \|d\| * \|d\| * (\|d\| + n))$

Table 1: Comparison of three kinds of 2D transformer in resource consumption. $batch$ indicates the batch size during training, $|d|$ indicates the number of clauses in the document, n refers to the hidden state size, w is equal to $2 * window + 1$, and $window$ is the window size used in window-constrained 2D transformer.

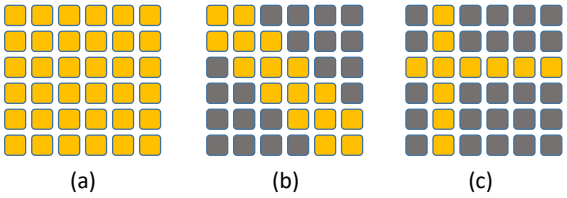


Figure 4: Examples of attentions to be calculated in three 2D Transformers: (a) Standard 2D-Transformer, (b) Window-constrained 2D Transformer, and (c) Cross-road 2D Transformer.

2.3.2 Window-constrained 2D Transformer

Considering that most of the cause clauses are around the emotion clauses, we propose the window-constrained 2D transformer, which is a standard 2D transformer while only takes c_i^{emo}-c_j^{cau} that meets $j - i \in [-window, window]$ as inputs.

The outputs of the window-constrained 2D transformer are $|d| * (window * 2 + 1)$ elements, each element requires the calculation of $|d| * (window * 2 + 1)$ attention weights, and eventually $(|d| * (window * 2 + 1)) * (|d| * (window * 2 + 1))$ weights are needed to be calculated and temporarily stored.

It should be noted that compared to the standard 2D transformer, the window-constrained transformer not only greatly reduces the resource requirements, but also alleviates the class imbalance problem to some extent since most of the pairs out of the windows are negative samples.

2.3.3 Cross-road 2D Transformer

Since the feature representation of pairs in the same row or column tends to be closer, we believe that pairs in the same row and column with the current pair have a greater impact on the current pair. Therefore, we propose the cross-road 2D transformer, in which the multi-head 2D self-attention mechanism is replaced by the cross-road 2D self-attention, and the other parts remain the same.

In the cross-road 2D self-attention, we calculate a set of row-wise weights $\beta_{i,j}^{\text{row}} = \{\beta_{i,j,1}^{\text{row}}, \beta_{i,j,2}^{\text{row}}, \cdots, \beta_{i,j,|d|)}^{\text{row}}\}$ and a set of column-wise weights $\beta_{i,j}^{\text{col}} = \{\beta_{i,j,1}^{\text{col}}, \beta_{i,j,2}^{\text{col}}, \cdots, \beta_{i,j,|d|)}^{\text{col}}\}$ for each pair c_i^{emo}-c_j^{cau}:

$$\beta_{i,j,b}^{\text{row}} = \frac{\exp(\frac{\boldsymbol{q}_{i,j} \cdot \boldsymbol{k}_{i,b}}{\sqrt{n}})}{\sum_{b'} \exp(\frac{\boldsymbol{q}_{i,j} \cdot \boldsymbol{k}_{i,b'}}{\sqrt{n}})}, \quad (14)$$

$$\beta_{i,j,a}^{\text{col}} = \frac{\exp(\frac{\boldsymbol{q}_{i,j} \cdot \boldsymbol{k}_{a,j}}{\sqrt{n}})}{\sum_{a'} \exp(\frac{\boldsymbol{q}_{i,j} \cdot \boldsymbol{k}_{a',j}}{\sqrt{n}})}. \quad (15)$$

Then the new feature representation of c_i^{emo}-c_j^{cau} is obtained by considering the pairs in the same row and column with it:

$$\hat{\mathbf{z}}_{i,j} = (\sum_{b=1}^{|d|} \beta_{i,j,b}^{\text{row}} \cdot \boldsymbol{v}_{i,b} + \sum_{a=1}^{|d|} \beta_{i,j,a}^{\text{col}} \cdot \boldsymbol{v}_{a,j})/2. \quad (16)$$

The outputs of the cross-road 2D transformer are $|d| * |d|$ elements, each element requires the calculation of $(|d| + |d|)$ attention weights, and eventually $(|d| * |d|) * (|d| * 2)$ weights are needed to be calculated and temporarily stored.

In this way, the new representation of each pair c_i^{emo}-c_j^{cau} can encode the information on all the pairs in the same row and column. In addition, if the cross-road 2D transformer is performed twice or more, the feature representation of each pair can encode the global information on all the pairs in $\boldsymbol{M}$, while standard 2D transformer requires much more resource to achieve this.

We show an example of attentions to be calculated for standard, window-constrained, and cross-road 2D transformer in Figure 4 (a), (b), and (c), respectively, and summarize their resource consumption in Table 1.

2.4 2D Emotion-Cause Pair Prediction

After a stack of N 2D transformer layers, we can get the final representation $\boldsymbol{o}_{i,j}^{(N)}$ for each pair c_i^{emo}-c_j^{cau}, and predict the emotion-cause pair distribution $\hat{\mathbf{y}}_{i,j}^{\text{pair}}$ as follows:

$$\hat{\mathbf{y}}_{i,j}^{\text{pair}} = \text{softmax}(W^{\text{pair}} o_{i,j}^{(N)} + b^{\text{pair}}). \quad (17)$$

The loss of emotion-cause pair classification for a document d is:

$$L^{\text{pair}} = -\sum_{i=1}^{|d|}\sum_{j=1}^{|d|} \mathbf{y}_{i,j}^{\text{pair}} \cdot \log(\hat{\mathbf{y}}_{i,j}^{\text{pair}}), \quad (18)$$

where $\mathbf{y}_{i,j}^{\text{pair}}$ is the ground truth distribution of emotion-cause pair of c_i^{emo}-c_j^{cau}.

In order to get better emotion-specific representation and cause-specific representation, we introduce the auxiliary loss for emotion prediction and cause prediction:

$$L^{\text{aux}} = -\sum_{i=1}^{|d|} \mathbf{y}_i^{\text{emo}} \cdot \log(\hat{\mathbf{y}}_i^{\text{emo}}) - \sum_{i=1}^{|d|} \mathbf{y}_i^{\text{cau}} \cdot \log(\hat{\mathbf{y}}_i^{\text{cau}}), \quad (19)$$

where $\mathbf{y}_i^{\text{emo}}$ and $\mathbf{y}_i^{\text{cau}}$ are emotion and cause annotation of clause c_i, respectively. The final loss of our model for a document d is a weighted sum of L^{pair} and L^{aux} with L2-regularization term as follows:

$$L = \lambda_1 L^{\text{pair}} + \lambda_2 L^{\text{aux}} + \lambda_3 ||\theta||^2, \quad (20)$$

where $\lambda_1, \lambda_2, \lambda_3 \in (0, 1)$ are weights, θ denotes all the parameters in this model.

3 Experiments

3.1 Dataset and Metrics

We evaluated our proposed model on an ECPE corpus from (Xia and Ding, 2019), which was constructed based on a Chinese emotion cause corpus (Gui et al., 2016a). The same as (Xia and Ding, 2019), we stochastically select 90% of the data as training data and the remaining 10% as testing data. In order to obtain statistically credible results, we repeat the experiments 20 times and report the average result. The precision, recall, and F1 score defined in (Xia and Ding, 2019) are used as the metrics for evaluation.

In addition, we also evaluated the performance of two sub-tasks: emotion extraction and cause extraction, using the precision, recall, and F1 score defined in (Gui et al., 2016a) as the metrics.

3.2 Experimental Settings

We use word vectors provided by (Xia and Ding, 2019) that were pre-trained on a corpora from Chinese Weibo. The dimensions of word embedding and relative position embedding are set to 200 and 50, respectively. The number of hidden units in BiLSTM for all our models is set to 100. The dimension of the hidden states, query, key, and value in the transformer are all set to 30. The window size in the window-constrained 2D transformer is set to 3. All weight matrixes and bias are randomly initialized by a uniform distribution U(0.01, 0.01).

For training details, we use the stochastic gradient descent (SGD) algorithm and Adam update rule with shuffled minibatch. The batch size and learning rate are set to 32 and 0.005, respectively. As for regularization, dropout is applied for word embeddings and the dropout rate is set to 0.7. The weights $\lambda_1, \lambda_2, \lambda_3$ in formula 20 are set to 1, 1, 1e-5, respectively. The code has been made publicly available on Github[1].

3.3 Overall Performance

Table 2 shows the experimental results of our models and baseline methods on the ECPE task as well as two subtasks (emotion extraction and cause extraction).

ECPE-2Steps is a set of two-step pipeline methods proposed in our prior work (Xia and Ding, 2019), which first perform individual emotion extraction and cause extraction via multi-task learning, and then conduct emotion-cause pairing and filtering. Specifically, there are three kinds of multi-task learning settings:

1) Indep: It is an independent multi-task learning method, in which emotion extraction and cause extraction are independently modeled.

2) Inter-CE: It is an interactive multi-task learning method, in which the predictions of cause extraction are used to improve emotion extraction.

3) Inter-EC: It is another interactive multi-task learning method, in which the predictions of emotion extraction are used to enhance cause extraction.

ECPE-2D is a joint framework proposed in this paper, which integrates the 2D emotion-cause pair representation, interaction, and prediction in an

[1] https://github.com/NUSTM/ECPE-2D

Framework	Approach		Emotion-Cause Pair Ext.			Emotion Ext.			Cause Ext.		
			P	R	$F1$	P	R	$F1$	P	R	$F1$
ECPE-2Steps	Indep		68.32	50.82	58.18	83.75	80.71	82.10	69.02	56.73	62.05
	Inter-CE		69.02	51.35	59.01	84.94	81.22	83.00	68.09	56.34	61.51
	Inter-EC		67.21	57.05	61.28	83.64	81.07	82.30	70.41	60.83	65.07
ECPE-2D (Ours)	Indep	-	71.60	55.95	62.63	**86.32**	81.52	83.80	69.15	59.72	63.97
		+WC	69.01	59.58	63.80	85.08	81.82	83.35	71.57	59.08	64.64
		+CR	69.12	58.78	63.38	85.27	81.82	83.44	69.73	59.37	63.99
	Inter-CE	-	69.35	57.24	62.61	86.12	82.40	84.16	69.77	59.42	63.98
		+WC	68.62	58.70	63.18	84.97	82.58	83.70	69.24	59.15	63.65
		+CR	69.22	59.04	63.56	84.82	82.88	83.76	69.80	58.78	63.68
	Inter-EC	-	71.73	57.54	63.66	85.37	81.97	83.54	71.51	62.74	66.76
		+WC	71.18	59.84	64.94	85.11	82.37	83.65	71.33	62.85	66.72
		+CR	69.60	61.18	64.96	85.12	82.20	83.58	72.72	62.98	67.38
	Inter-EC (BERT)	-	70.73	64.86	67.47	86.22	91.82	88.88	**73.46**	68.79	70.96
		+WC	**72.92**	65.44	**68.89**	86.27	92.21	**89.10**	73.36	**69.34**	**71.23**
		+CR	69.35	**67.85**	68.37	85.48	**92.44**	88.78	72.72	69.27	70.87

Table 2: Performance of our models and baseline models (Xia and Ding 2019) using precision, recall, and F1-measure as metrics on the ECPE task as well as the two sub-tasks.

end-to-end fashion. We explored three individual emotion/cause encoding settings: Indep, Inter-CE and Inter-EC, and three emotion-cause pair interaction settings:

1) "-" indicates that we do not introduce emotion-cause pair interaction;

2) "+WC" indicates that we use the window-constrained 2D transformer for emotion-cause pair interaction;

3) "+CR" indicates that we use the cross-road 2D transformer for emotion-cause pair interaction;

Note that due to the limitations of GPU memory, we have not been able to perform experiments with Standard 2D Transformer.

First of all, it can be seen that our proposed model ECPE-2D (Inter-EC+WC) performs better than ECPE-2Step on all metrics of all tasks, which proves the effectiveness of our method.

On the ECPE task, ECPE-2Steps (Inter-EC) performs best among all the previous methods. Compared with ECPE-2Steps (Indep), the improvement of ECPE-2Steps (Inter-EC) is mainly on the recall rate, while the precision score is slightly reduced. On the basis of ECPE-2Steps (Inter-EC), the recall rate of ECPE-2D (Inter-EC+CR) has been further greatly improved, and the precision score has also been slightly improved, which ultimately leads to better performance on the F1 score.

On the emotion extraction and cause extraction subtasks, ECPE-2Steps (Inter-CE) and ECPE-2Steps (Inter-EC) achieves significant improvements compared to ECPE-2Steps (Indep) on the former and latter subtask respectively by leveraging the interaction between emotion and cause. While our method ECPE-2D (Inter-EC+CR) outperforms the previous methods on both subtasks. We attribute the improvements to multi-task learning, as compared to the ECPE-2Steps (Inter-EC) model, ECPE-2D (Inter-EC+CR) additionally introduces the emotion-cause pair extraction task and trains the three tasks in a unified framework.

In addition, we also explored the effect of using BERT[2] (Devlin et al., 2019) as clause encoder in Inter-EC, which is denoted as Inter-EC (BERT). The experimental results in Table 2 show that the performance on all tasks can be further greatly improved (especially, the state-of-the-art F1 score on the ECPE task is improved from 61.28% to 68.89%) by adopting BERT as clause encoder.

3.4 ECPE-2D vs. ECPE-2Steps

In order to verify the effect of our proposed joint framework ECPE-2D, we discard the emotion-cause pair interaction module and compare ECPE-2D models with ECPE-2Step models based on the same individual encoding setting, the results are shown in Table 2.

By comparing ECPE-2D (Indep) with ECPE-2Step (Indep), we find that the performance of ECPE-2D (Indep) on all the metrics of all tasks (especially the ECPE task) are significantly improved. On the ECPE task, the performance of ECPE-2D (Indep) is even better than ECPE-2D

[2]BERT is only used to replace the word-level Bi-LSTM. Specifically, each clause in the document is feed into the BERT model independently, and the final hidden state of "[CLS]" is used as the clause representation. Our model is built based on this implementation: https://github.com/google-research/bert.

(Inter-EC), which is the prior state-of-the-art model. On the two subtasks, the performance has also been improved. We attribute the improvements to multi-task learning, as compared to the ECPE-2Step (Indep) model, ECPE-2D (Indep) additionally introduces the emotion-cause pair extraction task.

By comparing ECPE-2D (Inter-CE) and ECPE-2D (Inter-EC) with their two-step pipeline versions (ECPE-2Step (Inter-CE) and ECPE-2Step (Inter-EC)), we can draw similar conclusions. All these results prove that the proposed joint framework ECPE-2D is superior to the two-step pipeline framework ECPE-2Step in solving the ECPE task.

3.5 The Effectiveness of 2D Transformer

Comparing with the ECPE-2D (Indep) model, the ECPE-2D (Indep+WC/CR) models can achieve further improvement on the ECPE task, while the improvement on the two subtasks are not significant. Similar conclusions can be drawn when comparing ECPE-2D (Inter-CE) and ECPE-2D (Inter-CE+WC/CR) as well as ECPE-2D(Inter-EC) and ECPE-2D(Inter-EC+WC/CR). Particularly, compared to the strong baseline ECPE-2D (Inter-EC(BERT)), the performance can still be improved by introducing two kinds of 2D transformers. These results demonstrate that the window-constrained and cross-road 2D transformer can effectively improve the performance on the ECPE task via encoding interactive information between pairs.

In addition, we found that for ECPE-2D (Indep/Inter-CE/Inter-EC/Inter-EC(BERT)), the improvements brought by the introduction of window-constrained and cross-road 2D transformer are similar. These results indicate that the two 2D transformers are comparable.

3.6 The Effectiveness of Auxiliary Supervision

In order to explore the impact of the auxiliary supervision of two subtasks (emotion extraction and cause extraction) on the final performance of the ECPE task, we design the experiments in Table 3. "-AS" denotes the auxiliary supervision is removed (in practice, we set λ_2 in formula (20) to 0).

Compared with ECPE-2D (Indep/Inter-CE/Inter-EC), we find that the F1 score of ECPE-2D (Indep/Inter-CE/Inter-EC)-AS on the ECPE task decreased by about 1.4%, 2.2%, and 2.6%, respectively, which indicates that the supervisions of emotion extraction and cause extraction are important for the ECPE task. Nevertheless, the results of ECPE-2D (Indep)-AS are still better than ECPE-2Step (Indep) and comparable to the prior state-of-the-art result, which shows that emotion-cause pair extraction can be performed individually and proves the effectiveness of our joint framework.

	Emotion-Cause Pair Ext.		
	P	R	$F1$
Indep-AS	67.26	56.46	61.24
Indep+WC-AS	68.87	59.78	63.86
Indep+CR-AS	67.48	60.66	63.76
Inter-CE-AS	68.36	54.40	60.42
Inter-CE+WC-AS	67.12	60.79	63.44
Inter-CE+CR-AS	67.28	61.08	63.85
Inter-EC-AS	66.46	56.69	61.08
Inter-EC+WC-AS	67.79	60.47	63.81
Inter-EC+CR-AS	69.26	60.06	64.17

Table 3: Performance of our models on the ECPE task when the auxiliary supervisions of emotion extraction and cause extraction are removed. For brevity, the prefix "ECPE-2D" of all methods in this table are omitted.

Compared with ECPE-2D (Inter-EC+WC/+CR), the F1 score of ECPE-2D (Inter-EC+WC/+CR)-AS on the ECPE task decreased by about 1.1% and 0.8%, which is much less than the decrease between ECPE-2D (Inter-EC) and ECPE-2D (Inter-EC)-AS (drops 2.6%). These results lead to the conclusion that the negative impact of removing auxiliary supervision is reduced when pairwise encoders are introduced. From another perspective, when auxiliary supervisions are removed, the improvement brought by introducing pairwise encoders is greater. Comparing ECPE-2D (Inter-CE+WC/+CR), ECPE-2D (Indep+WC/+CR) and their "-AS" versions leads to similar conclusions. The above results again demonstrate the effectiveness of the proposed 2D transformer.

4 Related Work

The emotion-cause pair extraction (ECPE) task was first proposed in our prior work (Xia and Ding, 2019) and is derived from the traditional emotion cause extraction (ECE) task. Since the ECPE task was recently proposed, there is little work on it. We mainly introduce the related work of ECE task.

The emotion cause extraction (ECE) task was first proposed by Lee et al. (2010), with the goal to extract the word-level causes that lead to the given emotions in text. Based on the same task settings, there were some other individual studies that conducted ECE research on their own corpus us-

ing rule-based methods (Neviarouskaya and Aono, 2013; Li and Xu, 2014; Gao et al., 2015a,b; Yada et al., 2017) or machine learning methods (Ghazi et al., 2015; Song and Meng, 2015).

Based on the analysis of the corpus in (Lee et al., 2010), Chen et al. (2010) suggested that a clause may be the most appropriate unit to detect causes and transformed the task from word-level to clause-level. There was also some work based on this task setting (Russo et al., 2011; Gui et al., 2014). Recently, a Chinese emotion cause dataset was released by (Gui et al., 2016a,b; Xu et al., 2017), and has received much attention. Based on this corpus, a lot of traditional machine learning methods (Gui et al., 2016a,b; Xu et al., 2017) and deep learning methods (Gui et al., 2017; Li et al., 2018; Yu et al., 2019; Xu et al., 2019; Ding et al., 2019; Xia et al., 2019) were proposed.

In addition, there is also some work focused on cause detection for Chinese microblogs using a multiple-user structure and formalized two cause detection tasks for microblogs (current-subtweet-based cause detection and original-subtweet-based cause detection). (Cheng et al., 2017; Chen et al., 2018b,a).

The traditional ECE tasks suffer from two shortcomings: 1) the emotion must be annotated before cause extraction in ECE, which greatly limits its applications in real-world scenarios; 2) the way to first annotate emotion and then extract the cause ignores the fact that they are mutually indicative. To address this problem, we proposed the new emotion-cause pair extraction task in (Xia and Ding, 2019), which aims to extract the potential pairs of emotions and corresponding causes in a document. We have also proposed a two-step framework, which first extracts individual emotion set and cause set, and then pairs the corresponding emotions and causes. In this paper, we propose a new end-to-end approach to represent the emotion-cause pairs by a 2D representation scheme. Two kinds of 2D transformers, namely window-constrained and cross-road 2D transformers, are further proposed to model the interactions of different emotion-cause pairs. Finally, the 2D representation, interaction, and prediction are integrated into a joint framework.

5 Conclusions

The emotion-cause pair extraction (ECPE) task has drawn attention recently. However the previous approach employed a two-step pipeline framework and has some inherent flaws. In this paper, instead of a pipeline of two steps, we propose a joint end-to-end framework, called ECPE-2D, to represent the emotion-cause pairs by a 2D representation scheme, and integrate the 2D emotion-cause pair representation, interaction, and prediction into a joint a framework. We also develop two kinds of 2D Transformers, i.e., Window-constrained and Cross-road 2D Transformers, to further model the interaction of different emotion-cause pairs. The experimental results on the benchmark emotion cause corpus demonstrate that in addition to the advantages of joint modeling, our approach outperforms the state-of-the-art method by 7.6 percentage points in terms of the F1 score on the ECPE task.

Acknowledgments

We would like to thank three anonymous reviewers for their valuable comments. This work was supported by the Natural Science Foundation of China (No. 61672288). Zixiang Ding and Rui Xia contributed equally to this paper.

References

Ying Chen, Wenjun Hou, and Xiyao Cheng. 2018a. Hierarchical convolution neural network for emotion cause detection on microblogs. In *International Conference on Artificial Neural Networks (ICANN)*, pages 115–122.

Ying Chen, Wenjun Hou, Xiyao Cheng, and Shoushan Li. 2018b. Joint learning for emotion classification and emotion cause detection. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 646–651.

Ying Chen, Sophia Yat Mei Lee, Shoushan Li, and Chu-Ren Huang. 2010. Emotion cause detection with linguistic constructions. In *Computational Linguistics (COLING)*, pages 179–187.

Xiyao Cheng, Ying Chen, Bixiao Cheng, Shoushan Li, and Guodong Zhou. 2017. An emotion cause corpus for chinese microblogs with multiple-user structures. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 17(1):1–19.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*, pages 4171–4186.

Zixiang Ding, Huihui He, Mengran Zhang, and Rui Xia. 2019. From independent prediction to re-ordered prediction: Integrating relative position and global label information to emotion cause identification. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 6343–6350.

Kai Gao, Hua Xu, and Jiushuo Wang. 2015a. Emotion cause detection for chinese micro-blogs based on ecocc model. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, pages 3–14.

Kai Gao, Hua Xu, and Jiushuo Wang. 2015b. A rule-based approach to emotion cause detection for chinese micro-blogs. *Expert Systems with Applications*, 42(9):4517–4528.

Diman Ghazi, Diana Inkpen, and Stan Szpakowicz. 2015. Detecting emotion stimuli in emotion-bearing sentences. In *International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, pages 152–165.

Lin Gui, Jiannan Hu, Yulan He, Ruifeng Xu, Qin Lu, and Jiachen Du. 2017. A question answering approach to emotion cause extraction. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1593–1602.

Lin Gui, Dongyin Wu, Ruifeng Xu, Qin Lu, and Yu Zhou. 2016a. Event-driven emotion cause extraction with corpus construction. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1639–1649.

Lin Gui, Ruifeng Xu, Qin Lu, Dongyin Wu, and Yu Zhou. 2016b. Emotion cause extraction, a challenging task with corpus construction. In *Chinese National Conference on Social Media Processing*, pages 98–109.

Lin Gui, Li Yuan, Ruifeng Xu, Bin Liu, Qin Lu, and Yu Zhou. 2014. Emotion cause detection with linguistic construction in chinese weibo text. In *Natural Language Processing and Chinese Computing (NLPCC)*, pages 457–464.

Sophia Yat Mei Lee, Ying Chen, and Chu-Ren Huang. 2010. A text-driven rule-based system for emotion cause detection. In *NAACL HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 45–53.

Weiyuan Li and Hua Xu. 2014. Text-based emotion classification using emotion cause extraction. *Expert Systems with Applications*, 41(4):1742–1749.

Xiangju Li, Kaisong Song, Shi Feng, Daling Wang, and Yifei Zhang. 2018. A co-attention neural network model for emotion cause analysis with emotional context awareness. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 4752–4757.

Alena Neviarouskaya and Masaki Aono. 2013. Extracting causes of emotions from text. In *International Joint Conference on Natural Language Processing (IJCNLP)*, pages 932–936.

Irene Russo, Tommaso Caselli, Francesco Rubino, Ester Boldrini, and Patricio Martínez-Barco. 2011. Emocause: an easy-adaptable approach to emotion cause contexts. In *Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA)*, pages 153–160.

Shuangyong Song and Yao Meng. 2015. Detecting concept-level emotion cause in microblogging. In *World Wide Web (WWW)*, pages 119–120.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*, pages 5998–6008.

Rui Xia and Zixiang Ding. 2019. Emotion-cause pair extraction: A new task to emotion analysis in texts. In *Association for Computational Linguistics (ACL)*, pages 1003–1012.

Rui Xia, Mengran Zhang, and Zixiang Ding. 2019. RTHN: A RNN-transformer hierarchical network for emotion cause extraction. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 5285–5291.

Bo Xu, Hongfei Lin, Yuan Lin, Yufeng Diao, Liang Yang, and Kan Xu. 2019. Extracting emotion causes using learning to rank methods from an information retrieval perspective. *IEEE Access*, 7:15573–15583.

Ruifeng Xu, Jiannan Hu, Qin Lu, Dongyin Wu, and Lin Gui. 2017. An ensemble approach for emotion cause detection with event extraction and multi-kernel svms. *Tsinghua Science and Technology*, 22(6):646–659.

Shuntaro Yada, Kazushi Ikeda, Keiichiro Hoashi, and Kyo Kageura. 2017. A bootstrap method for automatic rule acquisition on emotion cause extraction. In *IEEE International Conference on Data Mining Workshops*, pages 414–421.

Xinyi Yu, Wenge Rong, Zhuo Zhang, Yuanxin Ouyang, and Zhang Xiong. 2019. Multiple level hierarchical network-based clause selection for emotion cause extraction. *IEEE Access*, 7(1):9071–9079.