Transition-based Directed Graph Construction for Emotion-Cause Pair Extraction

Chuang Fan[1], **Chaofa Yuan**[1], **Jiachen Du**[1], **Lin Gui**[2*], **Min Yang**[3], **Ruifeng Xu**[1,4,5*]
[1]Harbin Institute of Technology (Shenzhen), China [2]University of Warwick, UK
[3]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
[4]Joint Lab of Harbin Institute of Technology and RICOH, [5]Peng Cheng Laboratory
{fanchuanghit, bruceyuan123, jacobvan199165}@gmail.com
Lin.Gui@warwick.ac.uk, min.yang@siat.ac.cn, xuruifeng@hit.edu.cn

Abstract

Emotion-cause pair extraction aims to extract all potential pairs of emotions and corresponding causes from unannotated emotion text. Most existing methods are pipelined framework, which identifies emotions and extracts causes separately, leading to a drawback of error propagation. Towards this issue, we propose a transition-based model to transform the task into a procedure of parsing-like directed graph construction. The proposed model incrementally generates the directed graph with labeled edges based on a sequence of actions, from which we can recognize emotions with the corresponding causes simultaneously, thereby optimizing separate subtasks jointly and maximizing mutual benefits of tasks interdependently. Experimental results show that our approach achieves the best performance, outperforming the state-of-the-art methods by 6.71% ($p < 0.01$) in $F1$ measure.

1 Introduction

Emotion-cause pair extraction (ECPE) is a new task to identify emotions and the corresponding causes from unannotated emotion text (Xia and Ding, 2019). This involves several subtasks, including 1) Extracting pair components from input text, e.g., emotion detection and cause detection; 2) Combining all the elements of the two sets into emotion-cause pairs and eliminating the pairs that do not exist a causal relationship. For the former subtask, a clause can be categorized into "emotion", which usually contains an emotion keyword to express specific sentiment polarity, or "cause", which contains the reason or stimuli of an observed emotion. Then, the set of all possible emotion-cause pairs will be fed into the second subtask to determine the relationship. In general, it is an essential issue in emotion analysis since it provides a new perspective to investigate how emotions are provoked, expressed, and perceived.

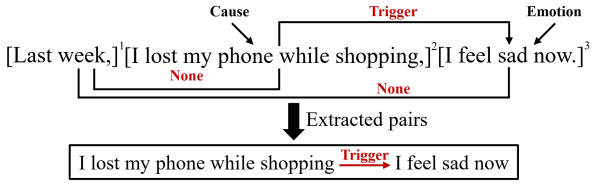


Figure 1: An example of emotion-cause pair extraction.

Figure 1 shows an example of ECPE, and the text is segmented into three clauses. In this instance, only the second clause and the third clause hold an emotion causality, where "*I lost my phone while shopping*" is the cause of emotion "*I feel sad now*". Thus, the extracted results of this sample should be {***I lost my phone while shopping***, ***I feel sad now***}. The goal of ECPE is to identify all the pairs that have emotion causality in an emotion text.

However, from both theoretical and computational perspectives, due to the inherent ambiguity and subtlety of emotions, it is hard for machines to build a mechanism for understanding emotion causality like human beings. Previous approaches mostly focused on detecting the causes towards the given annotation of emotions, which was followed by most of the recent studies in this field (Lee et al., 2010; Gui et al., 2014; Gao et al., 2015; Gui et al., 2016, 2017; Li et al., 2018; Xu et al., 2019; Fan et al., 2019). Nevertheless, it suffers that emotions must be annotated before extracting the causes, which limits the applications in real-world scenarios. Towards this issue, Xia and Ding (2019) presented a new task to extract emotion-cause pairs from the unannotated text. However, they followed a pipelined framework, which models emotions and causes separately, rather than joint decoding. Hence, to overcome the drawback of error propagation may occur in existing methods. Ideally,

* Co-Corresponding Authors

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 3707–3717

the emotion-cause structure should be considered as an integral framework, including representation learning, emotion-cause extraction, and reasoning.

To this end, we transform the ECPE problem into a procedure of directed graph construction, from which emotions and the corresponding causes can be extracted simultaneously based on the labeled edges. The directed graph is constructed by designing a novel transition-based parsing model, which incrementally creates the labeled edges according to the causal relationship between the connected nodes, through a sequence of defined actions. In this process, the emotion detection, cause detection, and their causality association can be jointly learned through joint decoding, without differentiating subtask structures, so that the maximum potential of information interaction between emotions and causes can be exploited. Besides, the proposed model processes the input sequence in a psycholinguistically motivated left to right order, consequently, reducing the number of potential pairs needed to be parsed and leading to speed up (if all clauses are connected by Cartesian products, the time complexity will be $O(n^2)$).

Regarding feature representation, BERT (Devlin et al., 2019) is used to produce the deep and contextualized representation for each clause, and LSTMs (Hochreiter and Schmidhuber, 1997) are performed to capture long-term dependencies among input sequences. In addition, action history and relative distance information between the emotion-cause pairs are also encoded to benefit the task.

To summarize, our contribution includes:

- Learning with a transition-based framework, so that end-to-end emotion-cause pair extraction can be easily transformed into a parsing-like directed graph construction task.
- With the proposed joint learning framework, our model can extract emotions with the corresponding causes simultaneously, often with linear time complexity.
- Performance evaluation shows that our model statistically significant improvements over the state-of-the-art methods on all the tasks[1].

2 Task Definition

The formal definition of emotion-cause pair extraction is given in (Xia and Ding, 2019). Briefly, given a piece of emotion text $d_1^n = (c_1, c_2, \ldots, c_n)$, which consists of several manually segmented clauses. The goal of ECPE is to output all potential pairs where exist emotion causality:

$$P = \{\cdots, (c^e, c^c), \cdots\} \quad (1)$$

where c^e is an emotion clause, and c^c is the corresponding cause clause.

Note that, the previous emotion cause extraction (ECE) task aims to extract c^c given the annotation of c^e: $\{c^c \rightarrow c^e\}$. In comparison, the ECPE is a new and challenging task since there is no annotation provided in the emotion text. Similar as the traditional ECE task, the ECPE is also defined at the clause level, because it is difficult to describe emotion causes at the word or phrase level. That is, in this paper, the "emotion" and "cause" are refer to "emotion clause" and "cause clause", respectively.

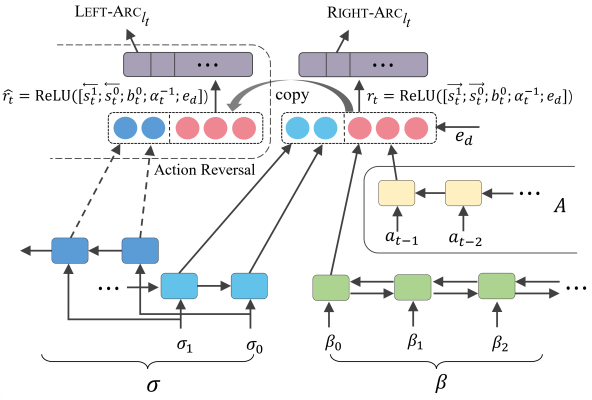


Figure 2: The architecture of our model. Dashed lines denote the components only work in training stage.

3 Our Approach

We present a new framework aimed at integrating the emotion-cause pair extraction into a procedure of parsing-like directed graph construction. The proposed framework incrementally constructs and labels the graph from input sequences, scoring partially segmented results using rich non-local features. Figure 2 shows the overall architecture of the proposed framework. In the following, we first introduce how to construct the directed graph based on a novel transition-based system, then the details of feature representation will be described.

3.1 Directed Graph Construction

Let $G = (V, R)$ be an edge-labeled directed graph where: $V = \{1, 2, \ldots, n\}$ is the set of nodes that correspond to clauses in the input text and

[1]The code and dataset are available at: https://github.com/HLT-HITSZ/TransECPE

Action	Change of state
SH	$\frac{(\sigma\|\sigma_1\|\sigma_0, \beta_0\|\beta, E, C, R)}{(\sigma\|\sigma_0\|\beta_0, \beta', E, C, R)}$
RA_{l_t}	$\frac{(\sigma\|\sigma_1\|\sigma_0, \beta_0\|\beta, E, C, R)}{(\sigma\|\sigma_0\|, \beta_0\|\beta, E \cup \{\sigma_0\}, C \cup \{\sigma_1\}, R \cup \{\sigma_1 \xrightarrow{l_t} \sigma_0\})}$
LA_{l_t}	$\frac{(\sigma\|\sigma_1\|\sigma_0, \beta_0\|\beta, E, C, R)}{(\sigma\|\sigma_1\|, \beta_0\|\beta, E \cup \{\sigma_1\}, C \cup \{\sigma_0\}, R \cup \{\sigma_1 \xleftarrow{l_t} \sigma_0\})}$
RA_{l_n}	$\frac{(\sigma\|\sigma_1\|\sigma_0, \beta_0\|\beta, E, C, R)}{(\sigma\|\sigma_0, \beta_0\|\beta, E \cup \{\sigma_0\}, C, R \cup \{\sigma_1 \xrightarrow{l_n} \sigma_0\})}$
LA_{l_n}	$\frac{(\sigma\|\sigma_1\|\sigma_0, \beta_0\|\beta, E, C, R)}{(\sigma\|\sigma_0\|\beta_0, \beta', E \cup \{\sigma_1\}, C, R \cup \{\sigma_1 \xleftarrow{l_n} \sigma_0\})}$
CA	$\frac{(\sigma\|\sigma_0, \beta_0\|\beta, E, C, R)}{(\sigma\|\sigma_0, \beta_0\|\beta, E \cup \{\sigma_0\}, C \cup \{\sigma_0\}, R \cup \{\sigma_0 \xrightarrow{l_t} \sigma_0\})}$

Table 1: Defined transition actions in our parser. For ease of illustration, we use the subscript $i \in \{0, 1, ...\}$ to denote the item index in the *stack* (starting from right), *buffer* and *action* (starting from left). That is, the top two items in the *stack* can be marked as $\sigma|\sigma_1|\sigma_0$ (similar to *buffer* and *action*).

$R = V \xrightarrow{R} V$ is the set of labeled edges. We will denote a connection between a head node $i \in V$ and a modifier node $j \in V$ as $i \xrightarrow{l} j$, where $l \in \{l_t, l_n\}$ is the causality label connecting them. l_t indicates the node i is the cause of the emotion node j while l_n indicates node j is an emotion but node i is not the corresponding cause. Besides, other nodes irrelevant to the final result have no edges. Note that, in this task, a node can be emotion and the corresponding cause simultaneously. Furthermore, an emotion node can also be associated with multiple causes. Thus, the acyclicity and single-head constraints are not necessary for our model, as arbitrary graphs are allowed.

We build the directed graph by designing a novel transition-based parser. Formally, each state of our parser is represented by a tuple: $S = (\sigma, \beta, E, C, R)$, where σ and β are disjoint lists called *stack* and *buffer*, which store the indices of nodes that have been processed and to be processed, respectively. E is the set of emotions, and C is the set of causes. R is used to store the edges generated so far. Besides, *action* history is stored to a list A.

The definition of action set plays a crucial role in the transition-based system, and it relies on the type of task. As shown in Table 1, we define 6 types of actions based on our empirical observation, and their logics are summarized as follows:

- SHIFT (SH). Pops β_0 and puts it on the top of σ. It is legal only when the β is not empty.

Stack	Buffer	Action	Emotion	Cause	Edge
[]	[1,2,3,$]	SH	$\varnothing$	$\varnothing$	$\varnothing$
[1]	[2,3,$]	SH	$\varnothing$	$\varnothing$	$\varnothing$
[1,2]	[3,$]	SH	$\varnothing$	$\varnothing$	$\varnothing$
[1,2,3]	[$]	RA_{l_t}	$\varnothing \cup \{3\}$	$\varnothing \cup \{2\}$	$2 \xrightarrow{l_t} 3$
[1,3]	[$]	RA_{l_n}	$\{3\} \cup \{3\}$	–	$1 \xrightarrow{l_n} 3$
[3]	[$]	SH	–	–	–
[3,$]	[]	–	–	–	–

Table 2: Transition sequence for the text in Figure 1.

- RIGHT-ARC$_{l_t}$ (RA_{l_t}). It assigns an edge from σ_1 to σ_0 with label l_t: $\sigma_1 \xrightarrow{l_t} \sigma_0$, then copies σ_0 to E and pops σ_1 from σ to C.
- LEFT-ARC$_{l_t}$ (LA_{l_t}). It assigns an edge from σ_0 to σ_1 with label l_t: $\sigma_1 \xleftarrow{l_t} \sigma_0$. Then copies σ_1 to E and pops σ_0 from σ to C
- RIGHT-ARC$_{l_n}$ (RA_{l_n}). Adds a relation from σ_1 to σ_0 with label l_n: $\sigma_1 \xrightarrow{l_n} \sigma_0$. Then pops σ_1 out of σ and only copies σ_0 to E.
- LEFT-ARC$_{l_n}$ (LA_{l_n}). It denotes a relation from σ_0 to σ_1: $\sigma_1 \xleftarrow{l_n} \sigma_0$ and copies σ_1 to E. Note that, we move β_0 to the top of σ to improve coverage rather than pops σ_0, because σ_0 may be the cause of incoming nodes in the β.
- CYCLE-ARC (CA). It assigns a loop edge on the node σ_0 with label l_t and then copies σ_0 to both E and C.

Action Constraints. To ensure that each parser state is valid, we need to specify some constraints on the action. For example, RIGHT-$*$ and LEFT-$*$ can only be conducted when there are at least two elements in the σ. We also empirically set a constraint that RIGHT-ARC$_{l_n}$ will be performed when $\sigma|\sigma_1|\sigma_0$ are both emotions but has no emotion causality. Additionally, in practical, CYCLE-ARC may conflict with other actions, e.g., σ_0 is the cause of itself but is also the cause of σ_1, which conflicts with the LEFT-ARC$_{l_t}$. For simplicity and efficiency, we separate it from other actions and distinguish it by training a binary classifier only depends on the representation of σ_0.

Table 2 illustrates the gold-standard sequence of transitions for the text in Figure 1. The parser state is initialized to $([\,], [1, 2, 3], \varnothing, \varnothing, \varnothing)$ and the terminal state is $([\ldots, \$], [\,], E, C, R)$, where \$ indicates the termination of transitions.

Search Algorithm. For the ECPE task, we transform it into a procedure of directed graph construction by a sequence of actions. The input is an emotion text $d_1^n = (c_1, c_2, \ldots, c_n)$ and the output is the corresponding sequence of actions $A_1^m = (a_1, a_2, \ldots, a_m)$. Hence, the task can be regarded as searching for an optimal action sequence A^* given the stream of clauses d_1^n:

$$A^* = argmax_A p(A_1^m | d_1^n) \tag{2}$$

Formally, at step t, our model predicts the next action based on the current system state S_t and the action history A_1^{t-1}. Thus, the task is modeled as:

$$(A^*, S^*) = argmax_{A,S} \prod_t p(a_t, S_{t+1} | A_1^{t-1}, S_t) \tag{3}$$

where a_t is the generated action at step t, and S_{t+1} is the updated system state according to a_t.

Let r_t to denote the representation for computing the probability of the action a_t at step t, this yields:

$$p(a_t | r_t) = \frac{exp(w_{a_t}^\top r_t + b_{a_t})}{\sum_{a' \in \mathcal{A}(S)} exp(w_{a'}^\top r_t + b_{a'})} \tag{4}$$

where w_a denotes a learnable parameter vector and b_a is a bias term. The set $\mathcal{A}(S)$ represents the legal actions that can be taken given the current parser state. Finally, the overall optimization function is:

$$\begin{aligned}(A^*, S^*) &= argmax_{A,S} \prod_t p(a_t, S_{t+1} | A_1^{t-1}, S_t) \\ &= argmax_{A,S} \prod_t p(a_t | r_t)\end{aligned} \tag{5}$$

where the ECPE is merged into a transition-based action prediction task. For efficient decoding, the maximum probability action is chosen greedily until the parsing procedure is termination.

3.2 Neural Transition-based Model

We apply BERT to produce the representation for each clause and use LSTMs to capture long-term dependencies of each parser state.

Representation of Clause. Given an emotion text $d_1^n = (c_1, c_2, \ldots, c_n)$ consisting of n clauses and each clause $c_i = (w_{i1}, w_{i2}, \ldots, w_{il})$ contains l words. We formulate each clause as a sequence $x_i = ([\text{CLS}], w_{i1}, \ldots, w_{il}, [\text{SEP}])$, where [CLS] is a special classification token that the final hidden state is used as the aggregate sequence features and [SEP] is a dummy token not used in our model. Thus, we obtain the hidden representation as $h_{c_i} = \text{BERT}(x_i) \in \mathbb{R}^{d_b * |x_i|}$ where d_b is the size of hidden dimension and $|x_i|$ is the length of sequence x_i. Then, the text d_1^n can be represented as $h_d = [h_{c_1}, h_{c_2}, \ldots, h_{c_n}]$.

Representation of Parser State. When the parsing starts, the parser state will be initialized to $([\,], [1, 2, \ldots, n], \varnothing, \varnothing, \varnothing)$ and a series of actions will consume the clauses in the *buffer* to incrementally build an output until reaches the terminal state $([\ldots, \$], [\,], E, C, R)$, as shown in Table 2.

Specifically, at step t, considering the triple (σ_t, β_t, A_t), where $\sigma_t = (\ldots, \sigma_1, \sigma_0)$, $\beta_t = (\beta_0, \beta_1, \ldots)$ and $A_t = (\ldots, a_{t-2}, a_{t-1})$. For the *stack*, to summarize the information from both directions, we use bidirectional LSTM to exploit two parallel passes, thus, the feature representation of σ_t is denoted as:

$$s_t = \text{LSTM}_\text{s}([\ldots, \overrightarrow{\sigma_1}, \overrightarrow{\sigma_0}], [\ldots, \overleftarrow{\sigma_1}, \overleftarrow{\sigma_0}]) \tag{6}$$

where $s_t = [\overrightarrow{s_t}, \overleftarrow{s_t}]$ that both $\overrightarrow{s_t}$ and $\overleftarrow{s_t} \in \mathbb{R}^{d_l * |\sigma_t|}$, d_l is the size of hidden dimension of LSTM and $|\sigma_t|$ is the size of σ_t . Similarly, we can get the representation for β_t by:

$$b_t = \text{LSTM}_\text{b}([\overrightarrow{\beta_0}, \overrightarrow{\beta_1}, \ldots], [\overleftarrow{\beta_0}, \overleftarrow{\beta_1}, \ldots]) \tag{7}$$

where $b_t = [\overrightarrow{b_t}, \overleftarrow{b_t}]$ that $\overrightarrow{b_t}$ and $\overleftarrow{b_t} \in \mathbb{R}^{d_l * |\beta_t|}$ where β_t is the size of β_t. For *action* sequence, we map each action a to a distributed representation e_a through a looking-up table E_a, and apply an unidirectional LSTM to obtain the complete history of actions from left-to-right:

$$\alpha_t = \text{LSTM}_\text{a}(\ldots, a_{t-2}, a_{t-1}) \tag{8}$$

Once a new action a_t is generated, the embedding e_{a_t} will be added into the rightmost position of the LSTM_a. To enhance the position relation between the pair (σ_1, σ_0), we also represent their relative distance d as an embedding e_d from a looking-up table E_d. The final representation of parser state at step t is the combination of these features.

Action Reversal. Let us visit the example in Figure 1 again. Reading it from left-to-right, as shown in the top of Figure 3, we see the clause "*I lost my phone while shopping*" trigger the emotion "*I feel sad now*", so the predicted action would be RIGHT-ARC$_{l_t}$. However, from a different perspective, we read it from right-to-left, as shown in the bottom

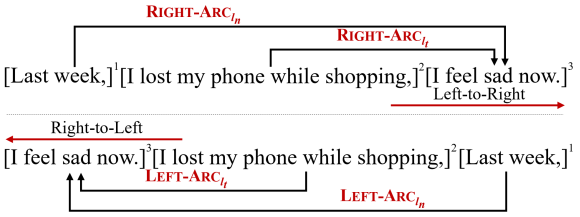


Figure 3: Illustration of action reversal.

of Figure 3, the cause "*I lost my phone while shopping*" behind the emotion "*I feel sad now*", so the predicted action should be reversed to LEFT-ARC$_{l_t}$. That is, $\overrightarrow{s_t}$ and $\overleftarrow{s_t}$ should be regarded as different features to produce different action. Based on this observation, we apply r_t and $\hat{r_t}$ to predict the original action and reversed action, respectively, which can be used to mine the deep directional information for this task:

$$r_t = \text{ReLU}([\overrightarrow{s_t}^1; \overrightarrow{s_t}^0; b_t^0; \alpha_t^{-1}; e_d]) \quad (9)$$

$$\hat{r_t} = \text{ReLU}([\overleftarrow{s_t}^1; \overleftarrow{s_t}^0; b_t^0; \alpha_t^{-1}; e_d]) \quad (10)$$

where ReLU is an activation function for nonlinearity. Index 0 and 1 indicate the first and second representation of σ and β, -1 indicates the last representation of action history.

Training. By learning with the transition-based framework, we convert the gold output structure in a set of training data into a gold sequence of defined actions. For each parser state at step t, we maximize the log-likelihood of the classifier in formula (5), which can be revised as:

$$\mathcal{J}(\theta) = \sum_t \log p(a_t|r_t) + \log p(\hat{a}_t|\hat{r}_t) + \log p(c_t|s_t^0) + \frac{\lambda}{2}||\theta||^2 \quad (11)$$

where $\hat{a}_t$ is the reversed action, and $p(c_t|s_t^0)$ is the predictive distribution of CYCLE-ARC which is separated from the other actions due to the action constraints. λ is the coefficient of L_2-norm regularization, and θ denotes all the parameters in this model. Note that, during the test decoding, only r_t and s_t^0 are used to predict the next action.

4 Dataset and Implementation Details

4.1 Dataset

To be consistent with previous approaches, we adopt the only benchmark (Gui et al., 2016) to evaluate our model by following (Xia and Ding, 2019). The corpus collected from SINA city news [2] and the details are summarized in Table 3.

Item	Num.	Item	Num.	Item	Num.
Emo_1	1816	Cau_1	1769	ECP_1	1746
Emo_2	118	Cau_2	156	ECP_2	177
Other	11	Other	20	Other	22

Table 3: Statistical information about the dataset. Emo_1 (Cau_1/ECP_1), Emo_2 (Cau_2/ECP_2) and other represent the texts with 1, 2 or more than 2 emotions (causes/emotion-cause-pairs).

4.2 Implementation Details

In this paper, we stochastically divide the corpus into a training/development/test set in a ratio of 8:1:1. In order to obtain statistically credible results, we evaluate our method 20 times with different data splits by following (Xia and Ding, 2019) and then perform one sample t-test on the experimental results. The average results of Precision (P), Recall (R) and F-measure ($F1$) are employed to measure the performance. Note that when we extract the emotion-cause pairs, we obtain the emotions and causes for each text simultaneously. Thus, we also evaluate the performance of emotion extraction and cause extraction in our model.

We adopt $\text{BERT}_{Chinese}$ as the basis in this work[3]. Adam optimizer is used for online learning (Kingma and Ba, 2015), and initial learning rates for the BERT layer and top MLP layer are set to 1e-5 and 1e-3, respectively. The hidden size of MLP layer is set to 256, and the hidden size of all LSTMs is set to 128 with 1 layer. The embeddings of position and action are initialized randomly with dimension 128 and keep unchanged during the training stage. The dropout rate is 0.5, the batch size is 3, and the coefficient of L_2 term is 1e-5. We train the model 10 epochs in total and adopt early stopping strategy based on the performance of development set. Then, the highest F-measure model on the development set is used to evaluate the test set.

5 Experiments

5.1 Baselines

We first compare our transition-based model with the method proposed by (Xia and Ding, 2019),

[2] http://news.sina.com.cn/society/

[3] Our BERT model is adapted from this implementation: https://github.com/huggingface/pytorch-pretrained-BERT

Method	Emotion extraction			Cause extraction			Emotion-cause pair extraction		
	P (%)	R (%)	$F1$ (%)	P (%)	R (%)	$F1$ (%)	P (%)	R (%)	$F1$ (%)
Indep	83.75	80.71	82.10	69.02	56.73	62.05	68.32	50.82	58.18
Inter-CE	84.94	81.22	83.00	68.09	56.34	61.51	69.02	51.35	59.01
Inter-EC	83.64	81.07	82.30	70.41	60.83	65.07	67.21	57.05	61.28
SL-BERT†	77.24	67.75	72.18	70.60	60.75	65.30	67.63	58.04	62.47
MT-BERT†	82.89	72.12	77.13	72.20	61.54	66.44	70.35	59.83	64.66
Ours†	**87.16**	82.44	**84.74**	**75.62**	64.71	**69.74**	**73.74**	63.07	**67.99**
LSTM$^{\dagger}_{based}$	80.80	**84.39**	82.56	67.42	**65.34**	66.36	65.15	**63.54**	64.34
-transition†	80.66	71.99	76.08	66.34	62.68	64.31	58.93	61.37	60.12

Table 4: Comparison with competitive baselines. † denotes the results are implemented in this paper. The results are average score over 20 runs, and the best scores are in bold.

which contains three models: 1) Indep: Emotion extraction and cause extraction are independently trained, then filtering the pairs that have no emotion causality; 2) Inter-CE: The difference is that the predictions of cause extraction are used to improve emotion extraction; 3) Inter-EC: Contrary to the Inter-CE, the predictions of emotion extraction are used to improve cause extraction. It is the current state-of-the-art model for this task.

To compare with other joint models, we implement SL-BERT (Zheng et al., 2017) and MT-BERT (Caruana, 1993) for this task. The former aims to joint extract entities and relations based on a novel tagging scheme with multiple labels and the other is a multi-task learning framework by sharing the hidden layers among all tasks. We implement them both based on BERT to be consistent with our experimental setting.

We also evaluate our model by only removing the transition procedure to reveal the effect of the transition-based algorithm, denoted as "-transition". Besides, for a fair comparison, we use LSTM as the basic encoder of clauses instead of BERT and keep the same experimental setting by following (Xia and Ding, 2019), namely LSTM$_{based}$.

5.2 Main Analysis

Table 4 shows the experimental results. With the transition-based algorithm, our proposed model achieves the best performance over all the three tasks, outperforming a number of competitive baselines by at least 1.74%, 3.30% and 3.33% in $F1$ score, respectively. The improvements are significant with $p < 0.01$ in one sample t-test.

Regarding pipelined approaches, Indep considers framework individually and ignores the fact that emotions and causes are usually mutually indicative, leading to the lowest performance. On the contrary, Inter-CE and Inter-EC yield better results by exploiting the relevance between emotions and causes. By comparing Inter-CE and Inter-EC, we find that the improvement of Inter-EC on cause extraction is much more than the improvement of Inter-CE on emotion extraction, thus Inter-EC shows better results. Differently, our model jointly extracts emotion-cause pairs and shows consistent performance improvement over the Indep-CE and Indep-EC, demonstrating the superiority of one-stage model by reducing error propagation.

In comparison with other joint models, our proposed model significantly outperforms SL-BERT by 12.56%, 4.44 % and 5.52% in $F1$ measure, respectively. We guess that SL-BERT jointly identifies emotion-cause pairs but still follows an emotion $\rightarrow$ cause pipelined decoding order. In contrast, we achieve fully joint decoding with interleaving actions for all the three tasks, thereby achieving better information interaction. Besides, our model also yields better results than MT-BERT, one possible reason is that the interdependence between the emotions and causes cannot be mined effectively only through parameter sharing.

We also show the results where BERT embeddings are replaced by LSTM from the input. It can be seen that the results still outperform the existing methods by at least 3.06% in $F1$ score. Furthermore, when we remove the transition procedure, the performance drops heavily over all the three tasks, especially with a 7.87% decrease in $F1$ measure on the ECPE task. These results show that the improvements provided by the proposed transition system are more noticeable than other components.

Method	Emotion extraction			Cause extraction			Emotion-cause pair extraction		
	P (%)	R (%)	$F1$ (%)	P (%)	R (%)	$F1$ (%)	P (%)	R (%)	$F1$ (%)
Ours	**87.16**	82.44	**84.74**	75.62	64.71	**69.74**	73.74	63.07	**67.99**
-reversal	85.26	83.63	84.43	**76.49**	63.08	69.14	74.59	61.35	67.33
-buffer	80.92	**86.94**	83.82	72.51	**65.65**	68.91	70.44	**63.73**	66.91
-action	82.18	86.69	84.34	**76.49**	61.69	68.30	**74.61**	60.04	66.53
-distance	81.60	85.05	83.29	75.93	57.89	65.69	74.06	56.29	63.96
-LSTM	81.23	83.37	83.29	72.19	59.20	65.06	70.64	57.82	63.59

Table 5: Feature ablation experiments. The results are average score over 20 runs, and the best scores are in bold.

5.3 Ablation Study

To further evaluate the contribution of neural components, we conduct feature ablation experiments to study the effects of different parts. As shown in Table 5, the $F1$ score decreases most heavily without *LSTM* (-4.40%), which indicates that it is necessary to capture non-local dependencies among input clauses, and our model can benefit from it effectively. *Distance* is also particularly relevant to the model by capturing the position information between the emotions and causes, which is consistent with our intuition that the closer a clause is to the emotion, the higher probability it should be the cause. Seen from the results, the history of actions stored in *action* has a crucial influence on predicting the next action. The results also show that *reversal*, which can be regarded as a data augmentation strategy, is useful by exploring the deep directional information. Without *buffer*, the $F1$ score drops 1.8% over the ECPE task. It may be due to the reason that *buffer* can provide more valuable information about the succeeding sequence.

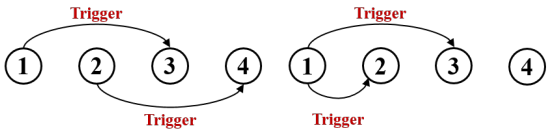
5.4 Action Set Validation

To gain more insights into the parsing procedure, we analyze the situations that emotion-cause pairs in an emotion text cannot be extracted entirely by our defined actions, as illustrated in Figure 4. For the pseudo sample in Figure 4(a), it can be parsed by the transition system using computation:

$$\mathrm{SH}(1);\mathrm{SH}(2);\mathrm{SH}(3);\mathrm{RA}_{l_n}(2 \xrightarrow{l_n} 3);$$
$$\mathrm{RA}_{l_t}(1 \xrightarrow{l_t} 3);\mathrm{SH}(4);\mathrm{RA}_{l_n}(3 \xrightarrow{l_n} 4);\mathrm{SH}(\$)$$

Similarity, for the pseudo sample in Figure 4(b), we get the transition sequence by:

$$\mathrm{SH}(1);\mathrm{SH}(2);\mathrm{RA}_{l_t}(1 \xrightarrow{l_t} 2);\mathrm{SH}(3);$$
$$\mathrm{RA}_{l_n}(2 \xrightarrow{l_n} 3);\mathrm{SH}(4);\mathrm{LA}_{l_n}(3 \xleftarrow{l_n} 4);\mathrm{SH}(\$)$$



(a) $(1 \xrightarrow{l_t} 3)$ and $(2 \xrightarrow{l_t} 4)$. (b) $(1 \xrightarrow{l_t} 2)$ and $(1 \xrightarrow{l_t} 3)$.

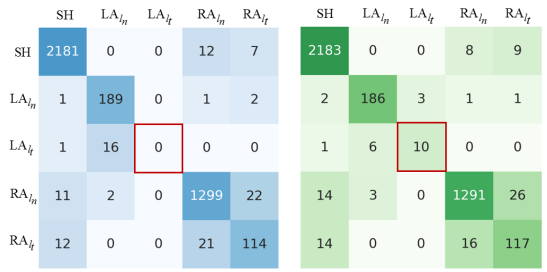
Figure 4: Pseudo samples that cannot be extracted entirely by our defined actions.

In both situations, our model can only extract one emotion-cause pair (i.e., $\mathrm{RA}_{l_t}(1 \xrightarrow{l_t} 3)$ and $\mathrm{RA}_{l_t}(1 \xrightarrow{l_t} 2)$, respectively.), because the cause which belongs to another emotion has been popped during the parsing procedure.

Based on this observation, one crucial problem about the proposed model is how many situations involving the emotion-cause transformation can be covered by the action set defined here. Although a formal theoretical proof is beyond the scope of this paper, we can empirically verify that the action set works well from Table 4. Going one step further, to further validate the actions, we input the texts into our transition system to obtain the "pseudo-gold" emotion-cause pairs P' based on the annotation, which can give us the correct action to take for a given parse state. Then we compare P' with the gold-standard emotion-cause pairs P to see how similar they are. On the whole dataset, we obtain an overall 98.5% $F1$ score for $\langle P, P' \rangle$, which indicates the upper bound of our transition system can achieve 98.5% in $F1$ score. Thus, the defined action set here is capable of extracting emotion-cause pairs through a sequence of actions.

5.5 Error Analysis

We also perform an experiment to understand the impact of action reversal on the performance. Fig-



(a) Without action reversal. (b) With action reversal.

Figure 5: Confusion matrices on test set. Vertical direction indicates the predicted action type and horizontal direction indicates the gold action type.

ure 5 shows the confusion matrices that present a comparison between the predicted actions and corrective actions. The results shows that SHIFT, LEFT-ARC$_{l_n}$ and RIGHT-ARC$_{l_n}$ yield higher accuracy on both Figure 5(a) and Figure 5(b) since they are account for a large proportion of the total actions. As expected, our model makes more mistakes involving the RIGHT-ARC$_{l_t}$ and LEFT-ARC$_{l_t}$, which play decisive roles in identifying the emotion-cause pairs. Especially for the LEFT-ARC$_{l_t}$ action, there is only about 0.43% in the total actions, turning out to be the most difficult action to learn given the relatively small training samples. Thus, as shown in Figure 5(a), the accuracy for LEFT-ARC$_{l_t}$ is 0, which drops the overall performance heavily. However, when we apply the action reversal into our model, boosting the accuracy of LEFT-ARC$_{l_t}$ by 58.8% and further improving the overall performance. We guess that based on action reversal, the original RIGHT-$*$ action can be reversed to LEFT-$*$ and vice versa, so that double the training actions. The results in Figure 5 show that our proposed model can capture this subtlety of emotions effectively by exploiting the deep directional information through action reversal strategy.

6 Related Work

Different from the traditional emotion analysis, which aims to identify emotion categories in text. Emotion cause extraction (ECE) reveals the essential information about what causes a certain emotion and why there is an emotional change. It is a more challenging task due to the inherent ambiguity and subtlety of emotion expressions.

Lee et al. (2010) first defined the emotion cause extraction as a word-level extraction task. They manually constructed a dataset from the Academia Sinica Balanced Chinese Corpus and generalized a series of linguistics rules based on the dataset. Based on this setting, there are some studies have been exploited for this task such as rule-based methods (Li and Xu, 2014; Gao et al., 2015; Yada et al., 2017) and machine learning methods (Ghazi et al., 2015; Song and Meng, 2015). Chen et al. (2010) converted the task from word-level to clause-level due to a clause may be the most appropriate unit to detect causes, and extracted causes using six groups of manually constructed linguistic cues. By following this task setting, Gui et al. (2014) extended the rule-based features to 25 linguistics cues, then trained classifiers on SVM and CRFs to detect causes. Gui et al. (2016) released a new Chinese emotion cause dataset collected from SINA city news [4] and proposed a multi-kernel based method to identify emotion causes. Following this corpus, Xu et al. (2019) proposed a learning to re-rank method based on a series of emotion-dependent and emotion-independent features. Recently, inspired by the success of deep learning architecture, some studies focused on identifying emotion causes with well designed neural network and attention mechanism (Gui et al., 2017; Li et al., 2018, 2019; Fan et al., 2019; Xia et al., 2019; Ding et al., 2019).

All of the above studies extracted emotion causes rely on the given emotion annotations, which limits the application in real-world scenarios due to the expensive annotations. Targeting this problem, Xia and Ding (2019) proposed a novel task based on ECE, namely emotion-cause pair extraction (ECPE), which aims at extracting emotions and the corresponding causes from unannotated emotion text. However, they followed a pipelined framework which first detects emotions and causes with individual learning frameworks, then performed emotion-cause pairing to eliminate the unmatched pairs, leading to a drawback of error propagation.

In this work, we design a novel transition-based model to extract emotions and causes simultaneously to maximize the mutual benefits of subtasks, thus alleviating the drawback of error propagation. Transition-based system is usually designed to model the chunk-level relation in a sentence for dependency parsing (Zhang and Nivre, 2011; Wang et al., 2015; Fernández-González and Gómez-Rodríguez, 2018). Apart from its application in dependency parsing, transition-based method has

[4] http://news.sina.com.cn/society/

also achieved great success in other natural language processing tasks, such as word segmentation (Zhang et al., 2016), information extraction (Wang et al., 2018b; Zhang et al., 2019), disfluency detection (Wang et al., 2017) and nested mention recognition (Wang et al., 2018a). To the best of our knowledge, this is the first work which extracts the emotion-cause pairs in an end-to-end manner.

7 Conclusion

In this paper, we present a novel transition-based framework to extract emotion-cause pairs as a procedure of directed graph construction. Instead of previous pipelined approaches, the proposed framework incrementally outputs the emotion-cause pairs as a single task, thereby the interdependence between emotions and causes can be exploited more effectively. Experimental results on a standard benchmark demonstrate the superiority and robustness of the proposed model compared to a number of competitive methods.

In the future, one possible direction is creating complete graphs with their nodes being input clauses to achieve full coverage. Besides, graph neural network-based (Kipf and Welling, 2016) methods are also worth investigating to model the relations among nodes for this task.

Acknowledgements

This work was partially supported by National Natural Science Foundation of China 61632011, 61876053, 61906185, Shenzhen Foundational Research Funding JCYJ20180507183527919, JCYJ20180507183608379, Key Technologies Research and Development Program of Shenzhen JSGG20170817140856618, EU-H2020 (grant no. 794196) and the project AWS13C008.

References

Rich Caruana. 1993. Multitask learning: A knowledge-based source of inductive bias. In *Machine Learning, Proceedings of the Tenth International Conference, University of Massachusetts, Amherst, MA, USA, June 27-29, 1993*, pages 41–48.

Ying Chen, Sophia Yat Mei Lee, Shoushan Li, and Chu-Ren Huang. 2010. Emotion cause detection with linguistic constructions. In *COLING 2010, 23rd International Conference on Computational Linguistics, Proceedings of the Conference, 23-27 August 2010, Beijing, China*, pages 179–187.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186.

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. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6343–6350. AAAI Press.

Chuang Fan, Hongyu Yan, Jiachen Du, Lin Gui, Lidong Bing, Min Yang, Ruifeng Xu, and Ruibin Mao. 2019. A knowledge regularized hierarchical approach for emotion cause analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5618–5628.

Daniel Fernández-González and Carlos Gómez-Rodríguez. 2018. Non-projective dependency parsing with non-local transitions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 693–700.

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

Diman Ghazi, Diana Inkpen, and Stan Szpakowicz. 2015. Detecting emotion stimuli in emotion-bearing sentences. In *Computational Linguistics and Intelligent Text Processing - 16th International Conference, CICLing 2015, Cairo, Egypt, April 14-20, 2015, Proceedings, Part II*, volume 9042 of *Lecture Notes in Computer Science*, pages 152–165. Springer.

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, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1593–1602.

Lin Gui, Dongyin Wu, Ruifeng Xu, Qin Lu, and Yu Zhou. 2016. Event-driven emotion cause extraction with corpus construction. In *Proceedings of the*

2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016, pages 1639–1649.

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 - Third CCF Conference, NLPCC 2014, Shenzhen, China, December 5-9, 2014. Proceedings*, pages 457–464.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

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. Association for Computational Linguistics.

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

Xiangju Li, Shi Feng, Daling Wang, and Yifei Zhang. 2019. Context-aware emotion cause analysis with multi-attention-based neural network. *Knowl.-Based Syst.*, 174:205–218.

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 *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4752–4757.

Shuangyong Song and Yao Meng. 2015. Detecting concept-level emotion cause in microblogging. In *Proceedings of the 24th International Conference on World Wide Web Companion, WWW 2015, Florence, Italy, May 18-22, 2015 - Companion Volume*, pages 119–120. ACM.

Bailin Wang, Wei Lu, Yu Wang, and Hongxia Jin. 2018a. A neural transition-based model for nested mention recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1011–1017.

Chuan Wang, Nianwen Xue, and Sameer Pradhan. 2015. A transition-based algorithm for AMR parsing. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 366–375.

Shaolei Wang, Wanxiang Che, Yue Zhang, Meishan Zhang, and Ting Liu. 2017. Transition-based disfluency detection using lstms. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2785–2794.

Shaolei Wang, Yue Zhang, Wanxiang Che, and Ting Liu. 2018b. Joint extraction of entities and relations based on a novel graph scheme. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4461–4467.

Rui Xia and Zixiang Ding. 2019. Emotion-cause pair extraction: A new task to emotion analysis in texts. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1003–1012.

Rui Xia, Mengran Zhang, and Zixiang Ding. 2019. RTHN: A rnn-transformer hierarchical network for emotion cause extraction. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5285–5291. ijcai.org.

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.

S. Yada, K. Ikeda, K. Hoashi, and K. Kageura. 2017. A bootstrap method for automatic rule acquisition on emotion cause extraction. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 414–421.

Junchi Zhang, Yanxia Qin, Yue Zhang, Mengchi Liu, and Donghong Ji. 2019. Extracting entities and events as a single task using a transition-based neural model. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5422–5428.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2016. Transition-based neural word segmentation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Yue Zhang and Joakim Nivre. 2011. Transition-based dependency parsing with rich non-local features. In

The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA - Short Papers, pages 188–193.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1227–1236.