



2017

## An Ensemble Approach for Emotion Cause Detection with Event Extraction and Multi-Kernel SVMs

Ruifeng Xu

*School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen Graduate School, Shenzhen 518055, China.*

Jiannan Hu

*School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen Graduate School, Shenzhen 518055, China.*

Qin Lu

*Department of Computing, the Hong Kong Polytechnic University, Hong Kong, China.*

Dongyin Wu

*School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen Graduate School, Shenzhen 518055, China.*

Lin Gui

*School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen Graduate School, Shenzhen 518055, China.*

Follow this and additional works at: <https://tsinghuauniversitypress.researchcommons.org/tsinghua-science-and-technology>



Part of the [Computer Sciences Commons](#), and the [Electrical and Computer Engineering Commons](#)

### Recommended Citation

Ruifeng Xu, Jiannan Hu, Qin Lu et al. An Ensemble Approach for Emotion Cause Detection with Event Extraction and Multi-Kernel SVMs. *Tsinghua Science and Technology* 2017, 22(6): 646-659.

This Research Article is brought to you for free and open access by Tsinghua University Press: Journals Publishing. It has been accepted for inclusion in *Tsinghua Science and Technology* by an authorized editor of Tsinghua University Press: Journals Publishing.

# An Ensemble Approach for Emotion Cause Detection with Event Extraction and Multi-Kernel SVMs

Ruifeng Xu, Jiannan Hu, Qin Lu, Dongyin Wu, and Lin Gui\*

**Abstract:** In this paper, we present a new challenging task for emotion analysis, namely emotion cause extraction. In this task, we focus on the detection of emotion cause a.k.a the reason or the stimulant of an emotion, rather than the regular emotion classification or emotion component extraction. Since there is no open dataset for this task available, we first designed and annotated an emotion cause dataset which follows the scheme of W3C Emotion Markup Language. We then present an emotion cause detection method by using event extraction framework, where a tree structure-based representation method is used to represent the events. Since the distribution of events is imbalanced in the training data, we propose an under-sampling-based bagging algorithm to solve this problem. Even with a limited training set, the proposed approach may still extract sufficient features for analysis by a bagging of multi-kernel based SVMs method. Evaluations show that our approach achieves an F-measure 7.04% higher than the state-of-the-art methods.

**Key words:** emotion cause detection; event extraction; multi-kernel SVMs; bagging

## 1 Introduction

In recent years, people can easily share experiences and emotions through the fast-growing Internet, anywhere and anytime. Since the rapid expansion of emotion information, how to analyze the emotions of an individual becomes a new challenge for Nature Language Processing (NLP). Recently, studies in emotion analysis<sup>[1-3]</sup> usually focus on emotion classification<sup>[4,5]</sup>, including detection of emotions expressed by writers of text<sup>[6]</sup>, predicting readers' emotions<sup>[7]</sup>, and so on. There are also some information extraction tasks in emotion analysis, such as extracting

the feeler of an emotion<sup>[8]</sup>, user modeling<sup>[9-11]</sup>, and imbalance data handling<sup>[12]</sup>. There are some other studies focused on joint learning with sentiments<sup>[13,14]</sup>, emotions in tweets or blogs<sup>[15-19]</sup>, and emotional lexicon construction<sup>[20-22]</sup>. However, these works all focus on the phenomenon of emotion expressions. In fact, we sometimes care more about the stimuli, or the cause of an emotion. For instance, as a reader, when we read text and see an emotional keyword like "exciting", we prefer to know the reason for the word "exciting" rather than the category of this emotion. Likely, manufacturers want to know why people love, or hate a certain product, because the cause of a customer's emotion is more useful to adjust the function of the product than the emotion itself. The White House also prefers to know the cause of emotional text "Let us hit the streets" rather than the distribution of different emotions.

Until now, there are still two main challenges in the study of emotion cause detection. First, the lack of open resources becomes the major limitation for researches in this area. It is so extremely expensive to construct a large-scale dataset for emotion cause detection that the size of the corpus for this task is usually very small.

---

• Ruifeng Xu, Jiannan Hu, Dongyin Wu, and Lin Gui are with School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen Graduate School, Shenzhen 518055, China. E-mail: xuruifeng@hit.edu.cn.

• Qin Lu is with Department of Computing, the Hong Kong Polytechnic University, Hong Kong, China. E-mail: csluqin@comp.polyu.edu.hk; guilin.nlp@gmail.com.

\* To whom correspondence should be addressed.

Manuscript received: 2017-01-02; accepted: 2017-03-26

Therefore, many machine-learning methods cannot be applied to this field. Another challenge is the absence of a formal definition on an event in the field of emotion cause detection. It leads to non-conforming metrics.

**Example 1** 当日, 跟中新网记者谈起建言献策的初衷, 白跃金陷入回忆, 并略显激动。

*On that day, talking about the original intention of giving advice, Yuejin Bai lost in memories, and seemed to be a little excited.*

The emotion is “激动/excited”. The cause of emotion is “陷入回忆/lost in memories”. The task we proposed aims to detect the cause of emotion in the text. The metric is on the clause level. It means that we need to identify which clause contains the cause of the emotion. In this example, the emotion cause is in the third clause “白跃金陷入回忆/Yuejin Bai lost in memories”.

In this paper, we first construct an annotated dataset for emotion cause extraction to be released to the public. To establish a prerequisite for the research, we provide a formal definition for the event. We then present a new emotion cause extraction method. The basic idea is to extract events in the context of emotional text through dependency parsing. Then, a 7-tuple representation structure is used to represent the nearby event. Based on this structured representation of the events, a multi-kernel SVM with polynomial kernel and tree kernel is used to determine whether an event expresses the emotion cause of an emotion or not. This method can detect almost all possible combinations of lexical structures to obtain sufficient features for emotion analysis using a limited training set. Since the distribution of events is unbalanced, we propose an ensemble learning method to solve this problem. Compared with existing methods, which used either manual rule sets or commonsense knowledge-bases to extend the information, our approach is machine-learning-based and still achieves state-of-the-art performance. The contributions of this work include both resources development and algorithm development.

The rest of this paper is organized as follows. Section 2 provides a review of related works on emotion analysis and emotion cause extraction. Section 3 presents the construction of emotion cause extraction corpus. Section 4 shows the method of event extraction. Section 5 provides the event-driven emotion cause extraction method and details the evaluations, as well as discussions. Section 6 concludes this work and recommends future directions.

## 2 Related Work

Emotion analysis is an essential subject in NLP and its applications<sup>[23]</sup>. However, most researches on emotion analysis focus on the emotion classification rather than on why we have emotions. In this section, we will introduce related works on the emotion analysis and emotion cause extraction.

The first issue in emotion analysis is to determine the taxonomy of emotions. Researchers have proposed a list of primary emotions<sup>[1-3]</sup>. In this study, we adopt Ekman’s emotion classification<sup>[2]</sup>, which identifies six primary emotions, namely happiness, sadness, fear, anger, disgust, and surprise, known as the “Big6” (<http://www.w3.org/TR/emotion-voc/xml#big6>) scheme in the W3C Emotion Markup Language. This list is agreed upon by most previous works in Chinese emotion analysis.

The second issue concerns how to perform emotion classification and emotion information extraction. Beck et al.<sup>[5]</sup> proposed a multi-task Gaussian-process based method for emotion classification. Gui et al.<sup>[4]</sup> used a coarse to fine method to classify emotions in Chinese blogs. Gao et al.<sup>[6]</sup> proposed a joint model to co-train a polarity classifier and an emotion classifier. Chang et al.<sup>[7]</sup> used a linguistic template to predict reader’s emotions. Das and Bandyopadhyay<sup>[8]</sup> used an unsupervised method to extract emotion feelers from Bengali blogs. There are other studies focused on joint learning with sentiments<sup>[13, 14]</sup>, emotions in tweets or blogs<sup>[15-19]</sup>, and emotional lexicon construction<sup>[20-22]</sup>. At the same time, sentiment analysis is also important in social networks<sup>[24, 25]</sup>. However, these related works all focused on the analysis of emotion expressions rather than the causes of the emotions.

Lee et al.<sup>[26]</sup> first presented work on emotion cause extraction. They manually annotated a small scale corpus from the Academia Sinica Balanced Chinese Corpus. Based on this corpus, they tried to extract the relationships between the changes from one emotion to another and the trigger factors. Then, they induced a series of clues of linguistics. On the basis of these works, Chen et al.<sup>[27]</sup> proposed an emotion cause extraction method based on template matching. Furthermore, there were some other studies<sup>[28-30]</sup> extending the rule-based method to informal text in Weibo (Chinese tweets). Gui et al.<sup>[28]</sup> and Li and Xu<sup>[29]</sup> used Lee et al.’s rules<sup>[26]</sup> to extract the emotion cause, such as Weibo text. Gao et al.<sup>[30]</sup> designed a

new method which is based on the OCC model. Other than the rule-based method, Ghazi et al.<sup>[31]</sup> applied CRFs to label the semantic roles on the emotion-related text. But this method requires that the emotion causes and their corresponding emotion keywords exist in the same sentence. By means of crowd-sourcing, Russo et al.<sup>[32]</sup> summarized some possible emotion cause phrases which were combined together randomly to obtain potential emotion causes based on an algorithm of co-occurrence frequency. More recently, our previous work<sup>[33]</sup> proposed an event-driven method to detect the emotion cause. However, for the unbalanced distribution of events, we only implemented an under-sampling method. It means that we removed some training data from the small training set. It is obviously detrimental to the performance. In summary, it is still challenging to extend the commonsense knowledge-base automatically.

For emotion cause extraction, from the above statement, we know that the resources used above are all small and not publicly accessible. Most of the methods are based on rules. Machine-learning methods, as well as deep-learning methods, are not active in this area, because of the lack of information in those small scale datasets. Furthermore, the existing methods lack an ability of understanding and generalizing the nature of emotion cause.

### 3 Construction of the Corpus

In this section, we first describe the linguistic phenomenon in emotion expressions. It inspires us to develop the annotated dataset. We then introduce details of the annotation scheme and the construction of the dataset, respectively.

#### 3.1 Linguistic phenomenon of emotion causes

Emotion cause plays an important role in an emotional expression. An emotion cause reveals the stimulus to an emotion. In written text, the cause of an emotion is usually expressed in the context of the emotion's keywords. Thus, finding appropriate context of emotional keywords in the annotation is a prerequisite to identify the causes. Finding the relationship between an emotion's cause and an emotion's keyword is the key to extracting the causes.

Other important features are the presence of conjunctions and prepositions. These cue words indicate the discourse information between the clauses. To utilize discourse information, the basic analysis unit

should be at the clause level rather than at the sentence level.

The genre of the text is also important. Studies show that in informal text, emotional expressions can have overlapping emotion cause and emotion target<sup>[28]</sup>. Thus, some causes are simply annotated as the target in informal text. Therefore, some studies even incorporate cause extraction with target identification to improve the performance. However, our focus is on the emotion cause identification. So, we use formal news text to avoid the potential mix up. To summarize, we follow three basic principles during construction: (1) keep the whole context of the emotion keywords; (2) the basic processing unit is at the clause level; and (3) use formal text.

#### 3.2 Collection and annotation

We first take 3-year (2013–2015) Chinese city news from NEWS SINA of 20 000 articles as the raw corpus. Based on a list of 10 259 Chinese primary emotion keywords (keywords for short)<sup>[34]</sup>, we extract 15 687 instances by keywords matching from the raw data. Here, we call the presence of an emotion keyword an instance in the corpus. For each matched keyword, we extract three preceding clauses and three following clauses as the context of an instance. If a sentence has more than 3 clauses in each direction, the context will include the rest of the sentence to provide the complete context. For simplicity, we omit the cross-paragraph contexts.

Note that the presence of keywords does not necessarily convey emotional information due to different possible reasons, such as negative polarity and sense ambiguity. For example, “幸福/ happiness” is an emotion keyword of positive polarity. It can also be the name of a song. In addition, the presence of emotion keywords does not necessarily guarantee the existence of emotional cause either.

After removing those irrelevant instances, there are still 2105 instances. For each emotional instance, two annotators manually annotate the emotion categories and the causes in the W3C Emotion Markup Language format. Here are two examples. Example 2 shows an annotated emotional sentence in the corpus, presented as simplified Chinese, followed by an English translation. To save space, we removed the xml format. The basic analysis unit is a clause. An emotion cause is marked as < cause >, and an emotion keyword is marked as < keywords >. Besides, emotion type,

POS, position, and the length of the annotation are also annotated in Emotion Markup Language format.

**Example 2** 当日, 跟中新网记者谈起建言献策的初衷, <cause POS="v" Dis="-1" >白跃金陷入回忆</cause>, 并略显<keywords type="happiness" >激动</keywords>。

*On that day, talking about the original intention of giving advice, <cause POS="v" Dis="-1" > Yuejin Bai lost in memories</cause>, and seemed to be a little <keywords type="happiness" > excited</keywords>.*

As we can see, the format of the corpus we construct is like that in Example 2. Here, Example 2 only contains one cause. However, one emotion keyword may have more than one corresponding emotion cause. Example 3 shows a sample of having two relevant causes for one emotion keyword. In our dataset, only 59 instances have two or more causes.

**Example 3** 他们要顾及龙龙养父的感受。李涛夫妇告诉记者, <cause POS = "v" Dis = "-2">知道龙龙现在还活着</cause>, 还<cause POS = "v" Dis = "-1">这么有出息</cause >, 真的是非常<keywords type = "happiness" >高兴</keywords >, 这件事是天大的喜事和好事。

*They need to take the feeling of Longlong's adoptive father into consideration. Tao Li and his wife told the reporter that they feel quite < keywords type = "happiness" > happy </keywords > because of < cause POS = "v" Dis = "1" > knowing that Longlong is alive now < /cause >, and < cause POS = "v" Dis = "2" > he is so outstanding < /cause >, this becomes a big rejoicing and good thing.*

### 3.3 Details of the dataset and its annotations

In our dataset, each instance has only one emotion keyword and at least one corresponding emotion cause. We ensure that the emotion keyword and its causes in the same instance are relevant. Table 1 lists the number of extracted instances, clauses, and emotion causes.

From Table 1, we can see that most instances contain only one emotion cause (97.2%), where only 2.6% instances have two emotion causes and 0.2% instances have three.

The distribution of emotion types is listed in Table 2. From Table 2, almost half of the emotions goes for "Happiness" and "Sadness".

The distribution of causes' positions is shown in Table 3, from which we can see that 78% of the emotion causes adjoin the emotion keywords at the clause

**Table 1** Details of the dataset.

Item	Number
Instance	2105
Clause	11 799
Emotion cause	2167
Doc with 1 cause	2046
Doc with 2 causes	56
Doc with 3 causes	3

**Table 2** Distribution of emotions.

Emotion	Number	Percentage (%)
Happiness	544	25.83
Sadness	567	26.94
Fear	379	18.00
Anger	302	14.35
Hate	225	10.69
Surprise	88	4.18

**Table 3** Cause position of emotions.

Position	Number	Percentage (%)
Previous 3 clauses	37	1.7
Previous 2 clauses	167	8.1
Previous 1 clause	1180	54.45
In the same clause	511	23.6
Next 1 clause	162	7.5
Next 2 clauses	48	2.2
Next 3 clauses	11	0.5
Other	42	1.9

level. Obviously, position is an important feature for emotion cause extraction. This motivates us to consider distance-based features for emotion cause extraction.

Table 4 shows the phrase types of the emotion causes. Verbs and verb phrases can form 93% of all the cause events. Thus, they are the focus of our learning algorithm.

In the annotation process, two annotators work independently. It is important to distinguish between clause level and phrase level in the procedure of the emotion cause annotation. The clause level is to label the clause that contains the emotion cause. The phrase level is to determine the boundary of an emotion cause. In the clause level, if two annotators disagree with each other, a third annotator serves as an arbitrator. In the phrase level, the broader boundary of these two will be

**Table 4** Distribution of POS tag.

POS	Number	Percentage (%)
Noun/noun phrase	147	6.78
Verb/verb phrase	2020	93.21

selected if they have the same annotation at the clause level. The kappa value for the clause level annotation is 0.9287. It means this type of annotation is very reliable.

### 4 An Ensemble Approach for Emotion Cause Detection

In this paper, we extract the emotion cause based on the event extraction and multi-kernel SVMs. We first extract the event based on the dependency-parsing result. Then, we represent the events in the tree format. The Event Tree (ET) which is the emotion cause will be marked as positive, otherwise negative. Since the size of the negative samples is much larger than that of the positive ones, we use an under-sampling method with bagging of classifier to ensure the balance of positive and negative examples. Then, we use this classifier to obtain the probability of emotion cause for each ET to produce a ranking list of candidate ETs. The ET with the highest-ranking score will serve as the cause event for the current instance. The framework of this approach is shown in Fig. 1.

#### 4.1 Extraction method of events

Since our approach of emotion cause detection is based on events extraction, the structure of event becomes the key point of our model. In this section, we present the method of event extraction.

##### 4.1.1 Definition of events

As mentioned above, verbs and verb phrases can construct almost all the emotion causes. So, we propose

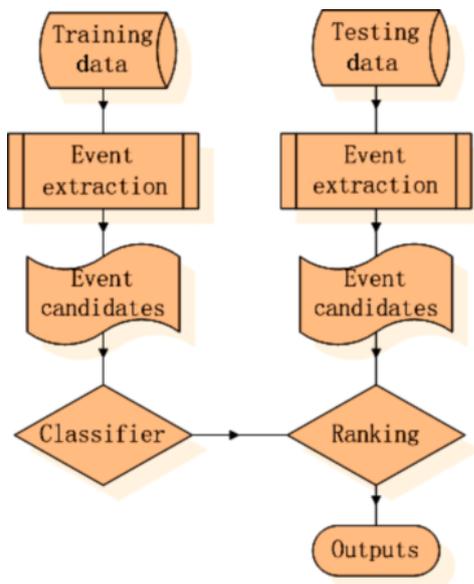


Fig. 1 The framework of our method.

a method which regards emotion causes as verb-based events. The basic idea is to use a type of event representation method to capture sufficient features from the raw data for emotion cause identification. So, it is important to formally define event first.

Inspired by the definition of an event in AI, for example, Radinsky and Davidovich<sup>[35]</sup> gave a formal definition of an event as an “action, actor, object, instrument, location, and time”. We used a similar definition for an event in our study. Actually, in the emotion cause extraction, the components of an event are simpler. We are only interested in the action, actor, and object (follow AI’s tradition, denote them as  $P$ ,  $O_1$ , and  $O_2$ , respectively). Since Chinese is an SVO language, the actor is the subject and the action is the verb. The subject and the object may have attributes and the predicate may have an adverb and a complement. These components may also be helpful in emotion cause extraction. To summarize, we formally define an emotion cause event as a 7-tuple:

$$e = (ATT_{O_1}, O_1, Adv, p, Cpl, ATT_{O_2}, O_2) \quad (1)$$

Here,  $ATT_{O_1}$  is the attribute of  $O_1$ , and  $ATT_{O_2}$  is the attribute of  $O_2$ . The Adv is an adverbial and Cpl is a complement. Note that in some cases, the syntactic components may be implicit, under this circumstance, the corresponding attributes can be filled with NIL values.

##### 4.1.2 Method of events extraction

To obtain the structure mentioned above, we first perform word segmentation, and then use the dependency parsing at the clause level to extract all possible relationships between any two relevant words. The dependency-parsing results have the form shown in Fig. 2 and Table 5.

Based on the dependency-parsing result, we can extract the event tuples from the raw data using the following method.



Fig. 2 A diagram of dependency parsing.

Table 5 The result of dependency parsing.

知道	龙龙	现在	还	活	着
0:HED	5:SBV	5:ADV	5:ADV	1:VOB	5:RAD

Since Chinese is a type of SVO language, the three basic components in Chinese are S (subject), V (verb), and O (object), which represents the actor, action, and object, respectively, in the definition of an event in our paper.

Now, we begin to construct the event tuples. First and foremost, we find the “SBV” (subject) string in the parsing result to obtain the subject.

Second, we search the following “VOB” (direct object), “IOB” (indirect object), or “FOB” (front object) in the parsing result to obtain the object. Then, we can get in the similar way. Now, we have two types of pre-events: verbs-subjects patterns and verbs-objects patterns. Because of the overlap between these two patterns, it is necessary to merge the patterns with the same verb.

Next, we should focus on the modifier of an event, such as the adverbial (marked as “Adv”) and complement (marked as “Cpl”) of the verbs, and the attributes (marked as “Att”) of the subjects or objects.

Finally, we can build up many event 7-tuples in the abovementioned form from the raw text. The pseudocode is shown in Algorithm 1.

It is obvious that not all clauses can be extracted from event tuples. In some cases, a clause may not contain a verb, let alone the subject and the object, because of their dependencies on the verb. However, in some other cases, one clause could contain a few verbs. In these circumstances, there can be several event tuples for different verbs in the same clause. In summary, one

---

#### Algorithm 1 Events extraction

---

```

1: Input: Clauses with dependency parsing result:
    $c_1, c_2, \dots, c_n$ .
2: Initialization: Event set  $E = \{\emptyset\}$ 
3: for all  $c_i, i \in \{1, 2, \dots, n\}$  do
4:   for all verb in  $c_i, \{v_j, j = 1, 2, \dots, k_i\}$  do
5:     Finding the subject of  $v_j$  by “SBV” relation
6:     Finding the object of  $v_j$  by “VOB”, “IOB” and
     “FOB” relations
7:     Finding the adverbial and complement of  $v_j$  by
     “Adv” and “Cpl” relations
8:     Finding the attribute of subject/object by “Att”
     relation
9:     For  $v_j$ , generate the event of 7-tuples based on
     formula (1):  $e_j$ 
10:     $E = E \cup \{e_j\}$ 
11:   end for
12: end for
13: Output: Event set  $E$ 

```

---

clause can produce  $n$  event tuples, here,  $n \geq 0$ . Though there may be no events in a clause, but a sentence must contain one or more events. Since in a sentence, there must be one or more verbs to maintain the complement of a sentence in the formal text. It is important to determine the difference between the clause level and the sentence level.

## 4.2 Event-driven emotion cause extraction

After the construction of Event Tuples (ETu), the issue of emotion cause extraction becomes a classification problem. If an ETu is extracted from an emotion cause, then this one is positive. Otherwise, it is negative. So, we need a binary classification algorithm to identify whether an ETu is an emotion cause or not. Due to an insufficient number of samples, it is necessary to provide more detailed information on these ETus. We choose a multi-kernel SVM with polynomial kernel and tree kernel because it can search almost all possible syntactic features under the multi-kernel-based function.

### 4.2.1 Event representation

According to the SVO structure of an ETu, we can construct an ET. We suppose that an ET has a fixed height of four levels. The first level is the ROOT node. Since Chinese is an SVO language, the descendant of the ROOT is S (subject), V (verb), and O (object). Based on this basic structure, an ET can be divided into three parts.  $(Att_{O_1}, O_1)$  belongs to S,  $(Adv, P, Cpl)$  belongs to V, and  $(Att_{O_2}, O_2)$  belongs to O. At last, we regard the words in every slot of the ET as leaves of the ET. Let us review Examples 2 and 3 again. We list two emotion cause clauses below with their corresponding ETs shown in Figs. 3–5.

(1) “白跃金陷入回忆/Yuejin Bai lost in memories”.

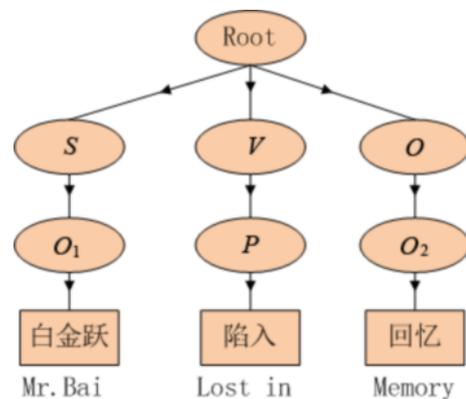


Fig. 3 The event tree for Example 2.

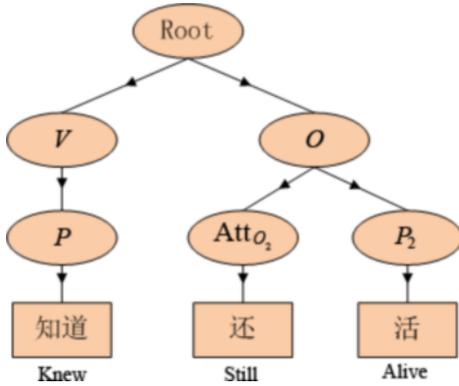


Fig. 4 One of the event trees for Example 3.

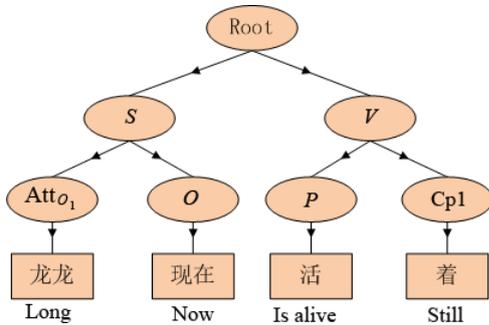


Fig. 5 The other event tree for Example 3.

(2) “知道龙龙现在还活着/knowing that Longlong is alive now”.

Here, since two verbs both exiting in the second example clause, we can construct two ETs from this clause. Based on this ET structure, we decide to use a tree kernel based SVMs to handle the ETs.

In addition, we need to add some lexical features to supplement the information needed in the task. According to the 7-tuple event structure, we can obtain the one-hot representation of each component as the feature. Assume that the representation of each is  $R_i$ , here  $i \in e$ , then we can capture the features of an ETu by a joint operation, called the ETu feature:

$$F = R_{Att_{o_1}} \oplus R_{O_1} \oplus \cdots \oplus R_{O_2} \quad (2)$$

Here,  $\oplus$  is the joint operation. It means combining the two representations into a new representation by order. Besides, as mentioned above, the distance between one component and the emotion keywords is so important that we take it as one of the features, with the joint operation, too. For the features, we decide to use a polynomial kernel function to make up the lexical features.

#### 4.2.2 Multi-kernel based function

As mentioned above, we extracted a type of tree

structure based feature, as well as some lexical features from the ETs. We need to combine these features through a type of classifier to dig the deeper information for detecting the underlying features for emotion cause extraction. Here, we choose a multi-kernel SVM with polynomial kernel and tree kernel to solve this problem.

**Multi-kernel function.** The multi-kernel function is designed with a predefined set of kernels and learns an optimal linear or non-linear combination of kernels as part of the algorithm. Instead of creating a new kernel, multi-kernel functions can be used to combine kernels already established for each individual set of data. In our method, we use the multi-kernel of a tree kernel based function with a polynomial kernel based function based SVMs to dig deeper information from ETs. For any two inputs  $T_1$  and  $T_2$  for the tree kernel function, with the respective features  $V_1$  and  $V_2$  for the polynomial kernel function, two types of multi-kernel functions are defined as follows:

$$K_{+multi}(T_1, T_2) = K_{tree}(T_1, T_2) + K_{vec}(V_1, V_2) \quad (3)$$

$$K_{*multi}(T_1, T_2) = K_{tree}(T_1, T_2) \times K_{vec}(V_1, V_2) \quad (4)$$

Here,  $K_{vec}$  refers to a polynomial kernel function.

Since our data can be represented in two different formats, we use two types of multi-kernels mentioned above for selecting an optimal kernel and parameters from a larger set of kernels, and for reducing bias due to kernel selection. Next, we introduce the polynomial kernel function and tree kernel function, respectively.

**Polynomial kernel function.** The polynomial kernel is one of the most widely used kernel functions for SVMs. Intuitively, the polynomial kernel looks not only at the given features of the input samples to determine their similarity, but also their combinations. The definition of the polynomial kernel is given as follows: for any two inputs  $V_1$  and  $V_2$ , where  $V$  represents the vector of features, the kernel function is defined as:

$$K_{polynomial}(V_1, V_2) = (\gamma V_1 \cdot V_2 + r)^d \quad (5)$$

Here,  $V_1$  and  $V_2$  are vectors in the feature space.  $\gamma$ ,  $r$ , and  $d$  are parameters. “ $\cdot$ ” is the inner product, where

$$V_1 \cdot V_2 = \sum_{i=1}^d v_{1i} \times v_{2i} \quad (6)$$

if  $V_1 = (v_{11}, v_{12}, \dots, v_{1n})$  and  $V_2 = (v_{21}, v_{22}, \dots, v_{2n})$ .

The polynomial kernel can represent the similarity of the vectors (in the training set) in the feature space over the polynomials of the original variables. We want to obtain more information from the original lexical features, so the polynomial kernel function is a good choice.

**Tree kernel function.** In NLP, it is often necessary to compare the tree structure for similarity. Such comparisons can be performed by computing the dot product of vectors of features of the trees. A well-designed tree kernel allows one to compute the similarity between trees without explicitly computing the feature vectors of these trees. For any two inputs  $T_1$  and  $T_2$  based on a tree structure, the kernel is defined as

$$K(T_1, T_2) = \sum_{n_1 \in T_1} \sum_{n_2 \in T_2} \delta(n_1, n_2) \quad (7)$$

Here  $n_1$  and  $n_2$  are tree nodes.  $\delta$  is a function defined recursively as: (1)  $\delta(n_1, n_2) = 0$  if the productions of  $n_1$  and  $n_2$  are different; (2) Else,  $\delta(n_1, n_2) = 1$  if  $n_1$  and  $n_2$  are matching in the pre-terminals; (3) Otherwise,  $\delta(n_1, n_2) = \prod_i (1 + \delta(c(n_1, i), c(n_2, i)))$ . Here,  $c(n, i)$  is the  $i$ -th node of  $n$ .

In our method, we construct sample tree structures from the ETs. To extract the internal and deep features from the event trees, we select the tree kernel based function.

### 4.2.3 Bagging of classifier

In our framework, we take the emotion cause relevant event as the positive sample. However, most of the events are non-relevant with emotion cause. It means that the distribution of training data is unbalanced. In our previous work<sup>[33]</sup>, we used a simple under-sampling method to randomly select part of the negative samples for training to keep training data balanced. In this method, the training data is devoid of full utilization. In this paper, we use a bagging method of the classifier to solve this problem. Essentially, we randomly select the negative samples several times to generate a different negative dataset. Then, we combine different negative datasets with the complete set of positive samples and train different classifiers to attain full utilization of the training data. The framework of our proposed method is shown in Fig. 6.

Here, for each classifier  $H_i$ , take an input sample as  $x_j$ , and the relevant label of  $x_j$  is  $y_j$ . Furthermore,  $y_j = 1$  indicates that  $x_j$  is an emotion cause event. Since we use the kernel based SVM as the basic classifier, the output of the classifier is the distance between the sample and the margin. So, we can get the likelihood conditional probability of  $x_j$  by sigmoid function:

$$P(x_j | y_j = 1) = \frac{1}{1 + e^{H_i(x_j)}} \quad (8)$$

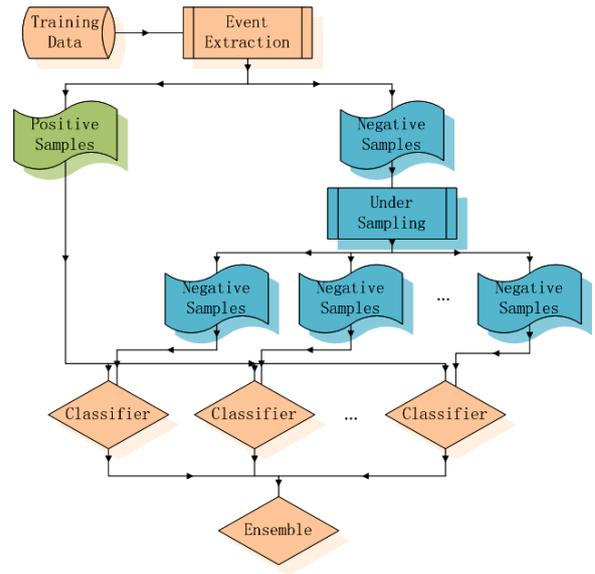


Fig. 6 The framework of bagging method for classification.

Then, the ensemble result of  $k$  classifiers should be

$$H(x_j) = \frac{1}{k} \sum_{i=1}^k \frac{1}{1 + e^{H_i(x_j)}} \quad (9)$$

Here, Eq. (9) is the average value of likelihood conditional probability. If  $H(x_j)$  is larger than 0.5, it means that  $x_j$  is an emotion cause event. Otherwise,  $x_j$  is not emotion cause. Furthermore, the higher  $H(x_j)$  indicates that  $x_j$  more likes an emotion cause event. So, Eq. (9) can be implemented in the ranking algorithm, such as the framework we proposed in Fig. 1.

## 5 Experiment

### 5.1 Experimental setup

In the experiments, we stochastically select 90% of the samples in the dataset as the training set and the remaining 10% as the test set. To obtain statistically credible results, we evaluate our methods and the reference methods 25 times, respectively. We conduct two groups of experiments. The first experiment evaluates the performance at the clause level to identify the clauses that contain emotion causes. The second experiment evaluates emotion causes through verb classification. This is because 93.21% of the emotion causes are verbs or verb phrases and verbs serve as the action component in our event definition. In particular, Gui et al.<sup>[33]</sup> conducted a set of similar experiments on the same set of data. However, they modified the tree kernel to bring the lexical words into a clause with

synonym information. In this work, we show that there is no need to modify any kernel function, using our event extraction method only with the original kernel function, we can achieve higher performance.

## 5.2 Emotion cause extraction

Since it is commonly accepted<sup>[29,30]</sup>, we decide to use the measurement proposed by Lee et al.<sup>[26]</sup> for emotion cause extraction. In this measurement, if the proposed emotion cause covers the annotated answer, the proposed sequence is considered correct. In this experiment, we compare our method with the following works:

- **RB** (Rule-based method): There are several studies regarding the RB method<sup>[26,28,29]</sup>. We use the union of the rules, and remove some rules which are not relevant to our dataset.
- **CB** (Commonsense-based method): In order to reproduce this method<sup>[32]</sup>, we use the Chinese Emotion Cognition Lexicon<sup>[36]</sup> as the commonsense. This lexicon resource contains more than 5000 emotion stimulations and their corresponding reflection words.
- **ML** (Rule-based features for machine learning): Use rules as features, and add other manual features for emotion cause classification<sup>[27]</sup>.
- **K<sub>vec</sub>**: Use the features from our previous work<sup>[37]</sup>, with original linear kernel function.
- **K<sub>word2vec</sub>**: Word2vec<sup>[37]</sup> is used to learn word representation. Use the representation according to our previous work<sup>[33]</sup> in the training of classifier.
- **K<sub>ET-O</sub>**: The original tree kernel used in our previous work<sup>[33]</sup>.
- **K<sub>ET-M</sub>**: The modified tree kernel used in our previous work<sup>[33]</sup>.
- **K<sub>new+O</sub>**: The multi-kernel of adding the polynomial kernel function with the original tree kernel in Ref. [33].
- **K<sub>new\*O</sub>**: The multi-kernel of multiplying the polynomial kernel and the original tree kernel in Ref. [33].
- **K<sub>new+M</sub>**: The multi-kernel of adding the polynomial kernel function with the modified tree kernel in Ref. [33].
- **K<sub>new\*M</sub>**: The multi-kernel of multiplying the polynomial kernel and the modified tree kernel of in Ref. [33].

- **B<sub>poly</sub>**: Only use bagging of polynomial kernel function with the lexicon features for emotion cause extraction.
- **B<sub>tree</sub>**: Only use bagging of tree kernel function with event trees to do emotion cause extraction.
- **B<sub>+multi</sub>**: Use bagging of a multi-kernel function by adding the polynomial kernel with the tree kernel function.
- **B<sub>\*multi</sub>**: Use bagging of a multi-kernel function by adding the polynomial kernel with the tree kernel function.

The performance of emotion cause detection at clause is given in Table 6. From Table 6, we can see that **B<sub>\*multi</sub>** achieves the top performance in F-measure. Compared to the other methods, the improvement is significant with a p-value less than 0.001 in the t-test.

Among the basic methods, RB achieves the top precision. However, its F-measure is limited by the low recall. Since CB is opposite to RB, the performance is improved when we use the output as the features to train a classifier in the RB+CB method. However, the improvement is quite limited, with F-measure = 0.0127. The F-measure of our reproduced RB is similar to other's results<sup>[28,29]</sup>. They repeated Lee et al.'s method<sup>[26]</sup> and achieved an F-measure of 0.55 more or less.

Chen et al.<sup>[27]</sup> reported that by using handcrafted rules as features to train a classifier with some additional features such as conjunction, action,

**Table 6 Performance of emotion cause detection at clause level.**

Method	Precision	Recall	F-measure
RB <sup>[26,28,29]</sup>	<b>0.6747</b>	0.4287	0.5243
CB <sup>[32,36]</sup>	0.2672	<b>0.7130</b>	0.3887
RB+CB <sup>[26,32,36]</sup>	0.5435	0.5307	0.5370
RB+CB+ML <sup>[26,27,36]</sup>	0.5921	0.5307	<b>0.5597</b>
K <sub>vec</sub> <sup>[33]</sup>	0.4200	0.4375	0.4285
K <sub>word2vec</sub> <sup>[33,37]</sup>	0.4301	0.4233	0.4136
K <sub>ET-O</sub> <sup>[33,37]</sup>	0.3982	0.4134	0.4057
K <sub>ET-M</sub> <sup>[33]</sup>	0.4583	0.4745	0.4662
K <sub>new+O</sub> <sup>[33]</sup>	0.6446	0.6779	0.6608
K <sub>new*O</sub> <sup>[33]</sup>	0.6492	0.6701	0.6595
K <sub>new+M</sub> <sup>[33]</sup>	0.6588	<b>0.6927</b>	0.6752
K <sub>new*M</sub> <sup>[33]</sup>	<b>0.6673</b>	0.6841	<b>0.6756</b>
B <sub>poly</sub>	0.6449	0.6284	0.6365
B <sub>tree</sub>	0.4329	0.4205	0.4266
B <sub>+multi</sub>	0.6482	0.6307	0.6393
B <sub>*multi</sub>	<b>0.7563</b>	<b>0.7352</b>	<b>0.7456</b>

and epistemic verbs, performance can be improved significantly. In our experiment, however, the result is contrary to this claim. The main reason is that the samples in Ref. [27] are less complex. About 85% of the emotion causes are in the same clause as the emotion keywords. Our corpus is quite different. The percentage of causes in the same clause with the emotion keywords is only about 23.6%. Chen et al.'s method<sup>[27]</sup> does not handle long-distance relations well. This explains why it does not work well for our dataset. Although (RB+CB+ML) does not perform well, there is still a 0.0354 improvement in the F-measure compared with RB. This means that the rules, combined with commonsense, are useful to obtain some underlying information for the machine-learning method.

In the work of Gui et al.<sup>[33]</sup>, as we see, the performance of the modified tree kernel for the F-measure in  $\mathbf{K}_{ET-M}$  is 0.0605 higher than that of the original tree kernel  $\mathbf{K}_{ET-O}$ . The consideration of terminal node really can significantly improve the performance of the tree kernel. The F-measure in the modified tree kernel  $\mathbf{K}_{ET-M}$  is also 0.0377 higher than  $\mathbf{K}_{vec}$  and 0.0526 higher than  $\mathbf{K}_{word2vec}$ . Here, we can see that kernel based syntactic representations have the ability of generalization. The original tree kernel without lexicon information underperforms compared to  $\mathbf{K}_{ET-M}$  but still obtains higher performance than both  $\mathbf{K}_{vec}$  and  $\mathbf{K}_{word2vec}$ . After combining the linear kernel function and the tree kernel function, the  $\mathbf{K}_{*multi}$  achieves an F-measure of 0.6756, which is higher than all other existing methods.

Among our proposed methods, we extract the events according to the way mentioned above first, and use the original polynomial kernel and tree kernel without any modification.  $\mathbf{B}_{poly}$  achieves an F-measure of 0.6365, which is 0.2080 higher than that of  $\mathbf{K}_{vec}$  and 0.2229 higher than that of  $\mathbf{K}_{word2vec}$ . This is an exciting improvement, which means that our method of extracting events is so effective that it can explore much more underlying features than that in the work of Gui et al.<sup>[33]</sup> Moreover,  $\mathbf{B}_{poly}$  also achieve an F-measure of 0.0768 higher than that of (RB+CB+ML). This shows that our method can dig more syntactic information than rules as well as commonsense. For the tree kernel, we only achieve an F-measure of 0.4266. Although lower than other performance of ours, it is still 0.0209 higher than  $\mathbf{K}_{ET-O}$ , which used the original tree kernel

with ETs in that work<sup>[33]</sup>. This result further testifies that our extraction method is much better, where  $\mathbf{B}_{+multi}$  obtained an F-measure of 0.6393. Compared to the basis methods, it achieves 0.0794 higher than (RB+CB+ML), which achieves the best result within the basis methods. At the same time, it is 0.0028 higher than  $\mathbf{B}_{poly}$  and 0.2127 higher than  $\mathbf{B}_{tree}$ . Despite little improvement from the original polynomial kernel to the multi(+) kernel, it can still reveal that the multi-kernel can benefit from both the polynomial kernel and the tree kernel. Therefore, we obtain many more features than any one-kernel based function. On the other hand, the multi(+) kernel can also work better than (RB+CB+ML), which just combines the rules and commonsense in a simple way. It is worth mentioning that  $\mathbf{B}_{*multi}$  obtained an F-measure of 0.7456, which is the highest performance among all the existing methods. This performance is 0.1063 higher than the multi(+) kernel, which means that the multi(\*) kernel can extract more useful internal features with multiply the polynomial kernel and the tree kernel. Compared to  $\mathbf{K}_{new*M}$ , which uses the modified tree kernel with the extracting method in our previous work<sup>[33]</sup>,  $\mathbf{B}_{*multi}$  achieves 0.0700 higher performance in F-measure. This means that our bagging method is so effective that any modification is not necessary. For the basis methods,  $\mathbf{B}_{*multi}$  can obtain 0.1859 higher performance in F-measure. The reason is that our method uses the event information which is at the syntactic level as well as the lexical level simultaneously. Full utilization of this information gives the model a generalization ability and can achieve better performance.

### 5.3 Verb classification for emotion cause

In our method, there are three types of representation levels: sentence, clause, and event tuple (ETu). The ETu is the most basic structure. Our classifier performs the classification operation on the ETus to decide whether the current ETu is emotion cause relevant through choosing the candidate cause events with the highest probability. The performance is measured by the verbs in the identified events. Here, we compared our methods with the basis methods as well as the work in Ref. [33]. The results are shown in Table 7.

In the experiment, we score every ETu. Here, each ETu has a unique id for its sentence id and for its clause id, since an ETu is extracted from a clause, and a clause comes from a sentence. This means that there

**Table 7 Performance on event trees classification.**

Method	Precision	Recall	F-measure
$\mathbf{K}_{\text{vec}}^{[33]}$	0.3500	0.2951	0.3192
$\mathbf{K}_{\text{word2vec}}^{[33,37]}$	0.3200	0.4833	0.3848
$\mathbf{K}_{\text{ET-O}}^{[33]}$	0.3906	0.2773	0.3228
$\mathbf{K}_{\text{ET-M}}^{[33]}$	0.3978	0.3303	0.3473
$\mathbf{K}_{\text{new+O}}^{[33]}$	0.4211	0.7219	0.5319
$\mathbf{K}_{\text{new*O}}^{[33]}$	0.4197	0.7305	0.5331
$\mathbf{K}_{\text{new+M}}^{[33]}$	0.4407	<b>0.7694</b>	0.5651
$\mathbf{K}_{\text{new*M}}^{[33]}$	<b>0.4532</b>	0.7504	<b>0.5646</b>
$\mathbf{B}_{\text{poly}}$	0.5801	0.4695	0.5188
$\mathbf{B}_{\text{tree}}$	0.3995	0.3168	0.3505
$\mathbf{B}_{\text{+multi}}$	0.5867	0.4703	0.5219
$\mathbf{B}_{\text{*multi}}$	<b>0.7071</b>	<b>0.5734</b>	<b>0.6330</b>

are many ETus in one sentence. Following this logic, we rank scores of ETus within the same sentence, and select the highest one. For this ETu, it is matched to a clause id with a unique sentence id. Furthermore, as mentioned above, there can be one or more than one ETus extracted from one clause. In other words, ETus and the corresponding clause must have a many-to-one relationship. So, we find that the whole performance in the experiment of verb classification is a little lower than that of emotion cause extraction at the clause level.

As we can see,  $\mathbf{B}_{\text{poly}}$  can achieve the highest performance in F-measure among  $\mathbf{K}_{\text{vec}}$ ,  $\mathbf{K}_{\text{word2vec}}$ , and itself, whereas  $\mathbf{B}_{\text{poly}}$  is 0.0138 lower than that of  $\mathbf{K}_{\text{word2vec}}$  in recall value. The reason is that the extraction result is based on ranking and only the top ranked event affects the performance. So, the precision is more important than the recall here. For the same reason,  $\mathbf{B}_{\text{tree}}$  achieves lower  $\mathbf{K}_{\text{ET-M}}$  in recall value and better F-measure value. When using  $\mathbf{B}_{\text{+multi}}$ , we get lower performance than all the multi-kernel based methods in Ref. [33]. This means that our event tuple structure is not very suitable for the original multi-kernel. However, when we use  $\mathbf{B}_{\text{*multi}}$ , the performance can reach 0.6330 in F-measure, the highest score among all the existing methods. This result shows that the multi-kernel using the multiply operation based SVMs not only provides a simple voting or joint method for the components, it also benefits from these two kernels to achieve better performance. Combined with our extraction method, the result can be optimal.

#### 5.4 Error analysis

Even with the best performance we can get, there are still mainly three types of errors in the results. These

include the following:

**(1) Emotion causes in other forms** In some cases, emotion causes may be present without verbs. For example:

**Example 4** 怕被人发现，为了保住脸面，  
<cause POS = “n” Dis = “-1”>情急之下不择手段  
</cause>，事后又<keywords type = “sadness”>  
追悔莫及</keywords>。

*Afraid of being found, as well as in order to be not too shame, he <cause POS = “n” Dis = “-1”>was careless of the consequences in haste </cause>. But too late to <keywords type = “sadness”>regret </keywords> at last.*

In this case, the emotion cause should be “情急之下不择手段/was careless of the consequences in haste”, which only contains two adverbial in it, not any verbs. But our method outputs “为了保住脸面/to be not too shame”, which contains an ETu in it. As an ETu-based method, our model cannot identify the correct emotion causes under these circumstances.

**(2) Objects of the emotion keywords** Emotion keywords can be verbs. In this case, the verbal emotion keyword may have an object, and the object of the emotion keyword can be the emotion cause in a large part. But in some cases, exceptions may happen as follows.

**Example 5** 以前有时候母亲会说妻子，但她都是一笑而过，<cause POS = “v” Dis = “-1”>  
两人基本没吵过嘴</cause>，所以母亲也特别  
<keywords type = “happiness”>喜欢</keywords>  
这个儿媳妇。

*My wife sometimes can be blamed by mum, but she has not care about it, and <cause POS = “v” Dis = “-1”>they rarely have quarrel </cause>. So, my mum <keywords type = “happiness”>like </keywords> this daughter-in-law very much.*

In this case, the reason why mum like the man’s wife is that “not quarrel”. But our method outputs “喜欢这个儿媳妇/this daughter-in-law”. The model takes this clause as an event tuple, which contains the object of “喜欢/like”. It ignores the real answer before the emotion keyword. Since the strong clue of the object of the emotion keywords, our method cannot tell this exception.

**(3) Errors caused by position** Usually, we choose the cause near the emotion keywords. But, sometimes, there are many clauses between the emotion keywords

and the real cause, which leads to errors.

**Example 6** 倪琼做出了令她至今<keywords type = “sadness”>愧疚不已</keywords>的事: 她背着不满周岁的孩子,去了医院附近的打印店,随后跑向闹市区,在福州总院公交站旁一连跪了4个小时,她手上的几张彩印传单很快引起了人们的关注,红字标题醒目地写着“求买孩子”,“因工伤老板逃跑,无钱医治, <cause POS = “v” Dis = “9”>愿将孩子卖了</cause>, 救救爸爸”。

*Qiong Ni has done something which made her feel <keywords type = “sadness” >guilty </keywords>. She run to the downtown area, and knelt for 4 hours at the bus stop, with a baby daughter on her back, after having gone to the print shop beside the hospital. What causes people’s attention is a few of colorful leaflets. The title of the red bulletins prominently reads “to sell the child”, “because of the boss’s escaping, no money to treat, <cause POS = “v” Dis = “9”>she is willing to sell her child </cause >, to save the father.”*

In this case, the emotion cause occurs at the position which is far away from the emotion keywords. This differs from the normal information we input into the model. Due to the small scale of the dataset, our model cannot identify this type of emotion cause.

In fact, parts of the errors can be attributed to our definition of the event tuple. When we decide to take an emotion cause as an event with the structure as mentioned in Eq. (1), we escape many emotion causes represented in other formats. But the result shows that our event tuple structure is effective. On the other hand, the position is also the main reason for these errors. Our model is weighted in favor of the clauses closest to the emotion keywords. However, the emotion cause may take place at the furthest clauses. There is much deeper information at the semantic level we need to dig, not only the simple and external one.

## 6 Conclusion

In this paper, we present our work about emotion cause extraction. There are two contributions in our paper. We first construct a corpus with annotated emotion cause from news text. We then propose an emotion cause extraction method based on an event extraction method. In this method, we use syntactic information to present the emotion cause in an event. Based on this type of event structure, a multi-kernel based method is designed to extract emotion causes. Compared with the baseline

method, which uses the manually constructed rules and commonsense knowledge-bases to detect the emotion cause, our proposed model can automatically obtain structural features and lexical features to achieve state-of-the-art performance on this dataset. In future work, we plan to extend this work into English and explore a bilingual method for this problem.

## Acknowledgment

This work was supported by the National Natural Science Foundation of China (Nos. 61370165, U1636103, and 61632011), Shenzhen Foundational Research Funding (Nos. JCYJ20150625142543470 and JCYJ20170307150024907), and Guangdong Provincial Engineering Technology Research Center for Data Science (No. 2016KF09).

## References

- [1] R. Plutchik, *Emotion: A Psychoevolutionary Synthesis*. New York, NY, USA: Harpercollins College Division, 1980.
- [2] P. Ekman, Expression and the nature of emotion, in *Approaches to Emotion*, K. Scherer and P. Ekman, eds. Hillsdale, NJ, USA: Erlbaum, 1984, pp. 19–344.
- [3] J. Turner, *On the Origins of Human Emotions: A Sociological Inquiry into the Evolution of Human Affect*. Stanford, CA, USA: Stanford University Press, 2000.
- [4] L. Gui, R. F. Xu, Q. Lu, J. C. Du, and Y. Zhou, Negative transfer detection in transductive transfer learning, *Int. J. Mach. Learn. Cybern.*, doi: 10.1007/s13042-016-0634-8.
- [5] D. Beck, T. Cohn, and L. Specia, Joint emotion analysis via multi-task Gaussian processes, in *Proc. 2014 Conf. Empirical Methods in Natural Language Processing*, Doha, Qatar, 2014, pp. 1798–1803.
- [6] W. Gao, S. S. Li, S. Y. M. Lee, G. D. Zhou, and C. R. Huang, Joint learning on sentiment and emotion classification, in *Proc. 22<sup>nd</sup> ACM Int. Conf. Information and Knowledge Management*, San Francisco, CA, USA, 2013, pp. 1505–1508.
- [7] Y. C. Chang, C. C. Chen, Y. L. Hsieh, C. C. Chen, and W. L. Hsu, Linguistic template extraction for recognizing reader-emotion and emotional resonance writing assistance, in *Proc. 53<sup>rd</sup> Annual Meeting of the Association for Computational Linguistics*, Beijing, China, 2015, pp. 775–780.
- [8] D. Das and S. Bandyopadhyay, Finding emotion holder from Bengali blog texts—An unsupervised syntactic approach, in *Proc. 24<sup>th</sup> Pacific Asia Conf. Language, Information and Computation*, Waseda, Japan, 2010, pp. 621–628.
- [9] L. Gui, R. F. Xu, Y. L. He, Q. Lu, and Z. Y. Wei, Intersubjectivity and sentiment: From language to knowledge, in *Proc. 25<sup>th</sup> Int. Joint Conf. Artificial Intelligence*, New York, NY, USA, 2016, pp. 2789–2795.

- [10] L. Gui, Y. Zhou, R. F. Xu, Y. L. He, and Q. Lu, Learning representations from heterogeneous network for sentiment classification of product reviews, *Knowl. Based Syst.*, vol. 124, pp. 34–45, 2017.
- [11] H. R. Xie, D. D. Wang, Y. H. Rao, T. L. Wong, L. Y. K. Raymond, L. Chen, and F. L. Wang, Incorporating user experience into critiquing-based recommender systems: A collaborative approach based on compound critiquing, *Int. J. Mach. Learn. Cybern.*, doi: 10.1007/s13042-016-0611-2.
- [12] S. J. Lin, Integrated artificial intelligence-based resizing strategy and multiple criteria decision making technique to form a management decision in an imbalanced environment, *Int. J. Mach. Learn. Cybern.*, doi: 10.1007/s13042-016-0574-3.
- [13] K. H. Luo, Z. H. Deng, L. C. Wei, and H. L. Yu, JEAM: A novel model for cross-domain sentiment classification based on emotion analysis, in *Proc. 2015 Conf. Empirical Methods in Natural Language Processing*, Lisbon, Portugal, 2015, pp. 2503–2508.
- [14] M. Mohtarami, M. Lan, and C. L. Tan, Probabilistic sense sentiment similarity through hidden emotions, in *Proc. 51<sup>st</sup> Annual Meeting on Association for Computational Linguistics*, Sofia, Bulgaria, 2013, pp. 983–992.
- [15] T. Hasegawa, N. Kaji, N. Yoshinaga, and M. Toyoda, Predicting and eliciting addressee’s emotion in online dialogue, in *Proc. 51<sup>st</sup> Annual Meeting on Association for Computational Linguistics*, Sofia, Bulgaria, 2013, pp. 964–972.
- [16] A. Qadir and E. M. Riloff, Learning emotion indicators from tweets: Hashtags, hashtag patterns, and phrases, in *Proc. 2014 Conf. Empirical Methods in Natural Language Processing*, Doha, Qatar, 2014, pp. 1203–1209.
- [17] G. Y. Ou, W. Chen, T. J. Wang, Z. Y. Wei, B. Y. Li, D. Q. Yang, and K. F. Wong, Exploiting community emotion for microblog event detection, in *Proc. 2014 Conf. Empirical Methods in Natural Language Processing*, Doha, Qatar, 2014, pp. 1159–1168.
- [18] H. H. Liu, S. S. Li, G. D. Zhou, C. R. Huang, and P. F. Li, Joint modeling of news reader’s and comment writer’s emotions, in *Proc. 51<sup>st</sup> Annual Meeting of the Association for Computational Linguistics*, Sofia, Bulgaria, 2013, pp. 511–515.
- [19] C. Q. Quan and F. J. Ren, Construction of a blog emotion corpus for Chinese emotional expression analysis, in *Proc. 2009 Conf. Empirical Methods in Natural Language Processing*, Singapore, 2009, pp. 1446–1454.
- [20] M. Yang, B. L. Peng, Z. Chen, D. J. Zhu, and K. P. Chow, A topic model for building fine-grained domain-specific emotion lexicon, in *Proc. 52<sup>nd</sup> Annual Meeting of the Association for Computational Linguistics (Short Papers)*, Baltimore, MD, USA, 2014.
- [21] J. Staiano and M. Guerini, Depechemood: A lexicon for emotion analysis from crowd-annotated news, arXiv preprint arXiv: 1405.1605, 2014.
- [22] S. M. Mohammad and P. D. Turney, Crowdsourcing a word-emotion association lexicon, *Comput. Intell.*, vol. 29, no. 3, pp. 436–465, 2013.
- [23] B. Liu, Sentiment analysis and opinion mining, in *Synthesis Lectures on Human Language Technologies*, G. Hirst, ed. San Rafael: Morgan & Claypool, 2012, pp. 1–167.
- [24] Y. Chen, Y. Chai, Y. Liu, and Y. Xu, Analysis of review helpfulness based on consumer perspective, *Tsinghua Sci. Technol.*, vol. 20, no. 3, pp. 293–305, 2015.
- [25] K. Tago and Q. Jin, Influence analysis of emotional behaviors and user relationships based on twitter data, *Tsinghua Sci. Technol.*, doi: 10.26599/TST.2018.9.10012.
- [26] S. Y. M. Lee, Y. Chen, and C. R. Huang, A text-driven rule-based system for emotion cause detection, in *Proc. NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, Los Angeles, CA, USA, 2010, pp. 45–53.
- [27] Y. Chen, S. Y. M. Lee, S. S. Li, and C. R. Huang, Emotion cause detection with linguistic constructions, in *Proc. 23<sup>rd</sup> Int. Conf. Computational Linguistics*, Beijing, China, 2010, pp. 179–187.
- [28] L. Gui, L. Yuan, R. F. Xu, B. Liu, Q. Lu, and Y. Zhou, Emotion cause detection with linguistic construction in Chinese Weibo text, in *Proc. 3<sup>rd</sup> Int. Conf. Natural Language Processing and Chinese Computing*, Shenzhen, China, 2014, pp. 457–464.
- [29] W. Y. Li and H. Xu, Text-based emotion classification using emotion cause extraction, *Expert Syst. Appl.*, vol. 41, no. 4, pp. 1742–1749, 2014.
- [30] K. Gao, H. Xu, and J. S. Wang, A rule-based approach to emotion cause detection for Chinese micro-blogs, *Expert Syst. Appl.*, vol. 42, no. 9, pp. 4517–4528, 2015.
- [31] D. Ghazi, D. Inkpen, and S. Szpakowicz, Detecting emotion stimuli in emotion-bearing sentences, in *Proc. 2015 Int. Conf. Intelligent Text Processing and Computational Linguistics*, Cairo, Egypt, 2015, pp. 152–165.
- [32] I. Russo, T. Caselli, F. Rubino, E. Boldrini, and P. Martínez-Barco, Emocause: An easy-adaptable approach to emotion cause contexts, in *Proc. 2<sup>nd</sup> Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, Portland, OR, USA, 2011, pp. 153–160.
- [33] L. Gui, D. Y. Wu, R. F. Xu, Q. Lu, and Y. Zhou, Event-driven emotion cause extraction with corpus construction, in *Proc. 2016 Conf. Empirical Methods in Natural Language Processing*, Austin, TX, USA, 2016, pp. 1639–1649.
- [34] L. H. Xu, H. F. Lin, Y. Pan, H. Ren, and J. M. Chen, Constructing the affective lexicon ontology, *J. China Soc. Sci. Tech. Inf.*, vol. 27, no. 2, pp. 180–185, 2008.
- [35] K. Radinsky and S. Davidovich, Learning to predict from textual data, *J. Artif. Intell. Res.*, vol. 45, no. 1, pp. 641–684, 2012.

[36] R. F. Xu, C. T. Zou, Y. Z. Zheng, J. Xu, L. Gui, B. Liu, and X. L. Wang, A new emotion dictionary based on the distinguish of emotion expression and emotion cognition, (in Chinese), *J. Chin. Inf. Process.*, vol. 27, no. 6, pp. 82–89, 2013.

[37] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, Distributed representations of words and phrases and their compositionality, in *Proc. 26<sup>th</sup> Int. Conf. Neural Information Processing Systems*, Lake Tahoe, NV, USA, 2013, pp. 3111–3119.



**Ruifeng Xu** is a professor and doctoral supervisor at Harbin Institute of Technology, Shenzhen Graduate School. He received the BEng degree from Harbin Institute of Technology, and MPhil and PhD degrees from the Hong Kong Polytechnic University, in 1995, 2001, and 2006, respectively. His research interests

include nature language processing, emotion computing, and text mining.



**Qin Lu** is a professor and doctoral supervisor in the Hong Kong Polytechnic University. She received the bachelor degree from Beijing Normal University, and MSc and PhD degrees from University of Illinois at Urbana-Champaign, in 1982, 1984, and 1988, respectively. Her research interests include natural language

processing, emotion computing, and ontology.



**Jiannan Hu** is currently a master student at Harbin Institute of Technology, Shenzhen Graduate School. She received the BA degree from Northeast Normal University in 2015. Her research interests include emotion understanding and user profiling.



**Lin Gui** received the PhD degree from Harbin Institute of Technology in 2017. His research interests include nature language processing, emotion computing, and machine learning.



**Dongyin Wu** received the master degree from Harbin Institute of Technology in 2016. She is currently an engineer in ByteDance Company. Her research interests include emotion analysis and corpus linguistics.