applied sciences

MDPI

Article

An End-to-End Mutually Interactive Emotion–Cause Pair Extractor via Soft Sharing

Beilun Wang [1,2,*], Tianyi Ma [3], Zhengxuan Lu [1] and Haoqing Xu [1]

[1] School of Computer Science and Engineering, Southeast University, Nanjing 211189, China
[2] Key Laboratory of Computer Network and Information Integration, Southeast University, Ministry of Education, Nanjing 211189, China
[3] School of Artificial Intelligence, Southeast University, Nanjing 211189, China
* Correspondence: beilun@seu.edu.cn

Abstract: Emotion–cause pair extraction (ECPE), i.e., extracting pairs of emotions and corresponding causes from text, has recently attracted a lot of research interest. However, current ECPE models face two problems: (1) The common two-stage pipeline causes the error to be accumulated. (2) Ignoring the mutual connection between the extraction and pairing of emotion and cause limits the performance. In this paper, we propose a novel end-to-end mutually interactive emotion–cause pair extractor (Emiece) that is able to effectively extract emotion–cause pairs from all potential clause pairs. Specifically, we design two soft-shared clause-level encoders in an end-to-end deep model to measure the weighted probability of being a potential emotion–cause pair. Experiments on standard ECPE datasets show that Emiece achieves drastic improvements over the original two-step ECPE model and other end-to-end models in the extraction of major emotional cause pairs. The effectiveness of soft sharing and the applicability of the Emiece framework are further demonstrated by ablation experiments.

Keywords: emotion–cause pair extraction; soft sharing; end-to-end model; multi-task learning; emotion–cause extraction

check for updates

Citation: Wang, B.; Ma, T.; Lu, Z.; Xu, H. An End-to-End Mutually Interactive Emotion–Cause Pair Extractor via Soft Sharing. *Appl. Sci.* **2022**, *12*, 8998. https://doi.org/10.3390/app12188998

Academic Editor: Valentino Santucci

Received: 27 July 2022
Accepted: 6 September 2022
Published: 7 September 2022

1. Introduction

Recently, emotion–cause extraction (ECE) has gained great popularity in text analysis [1,2]. ECE aims at extracting potential causes that lead to emotional expressions in the text. Instead of using word-level labeled sequence, ECE concentrates on the clause-level sequence, thus fully exploiting the linked relationship between different sentences [3]. This kind of clause-level observation improves the reliability of the ECE analysis greatly. In this respect, Ref. [4] first released a corresponding corpus, which was widely used in the following studies [5–11]. There are, however, two limitations [12] associated with the ECE task. On the one hand, ECE relies on the annotated text sentiment as the input at inference time, which limits its application. On the other hand, ECE ignores the mutual relationship between emotion and cause sentences in the text.

To solve the existing problems in ECE, Ref. [12] proposed a new task, emotion–cause pair extraction (ECPE), which aims to extract all possible emotion–cause pairs in the text without a given annotated sentiment word. Figure 1 illustrates the goal of ECPE and the key differences in comparison with ECE in the solution process. For the input document, the ECE task first identifies and annotates the emotion of the clauses, and then extracts the cause based on the annotated emotion. In the example shown in the figure, the emotion "*anger*" is first marked according to clause 7 "Jobs threw a tantrum" (Figure 1), and then the corresponding cause clause, "Scott assigned No.1 to Wozniak and No.2 to Jobs" (Figure 1), is extracted from the input document according to the emotion annotation. The ECPE task, in contrast to the ECE task, does not require annotations of sentiment from emotion clauses but notes that emotion and cause are mutually indicative. Therefore, all possible pairs are matched and filtered, and the two valid emotional cause pairs in this input document are directly derived:

(clause 7–clause 1) and (clause 8–clause 1), i.e., the emotion clause “Jobs threw a tantrum” (Figure 1) and its corresponding cause clause “Scott assigned No.1 to Wozniak and No.2 to Jobs” (Figure 1), and the emotion clause “even cried” (Figure 1) and its corresponding cause clause “Scott assigned No.1 to Wozniak and No.2 to Jobs” (Figure 1), without relying on the emotion annotations “*anger*” and “*sadness*” in the clause. The matching of two different emotion clauses with one identical cause clause, or the pairing of one emotion clause with two different cause clauses, as in the example in [12], reflects the increased attention of the ECPE task to the connection between the emotion and cause.

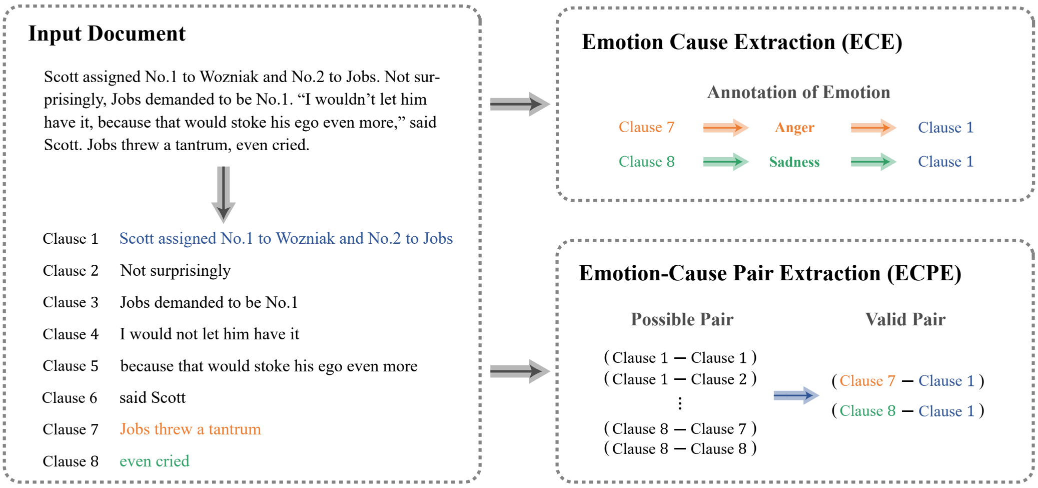


Figure 1. The difference between ECE tasks and ECPE tasks. ECE task aims to extract each cause clause provided emotion annotation, while the ECPE task is targeted at extracting all valid pairs of emotion clauses and the corresponding cause clause in an input document. The orange and green parts are emotion clauses, and the blue is a cause clause.

Currently, ECPE methods can be divided into two categories: two-step ECPE and end-to-end ECPE. The two-step ECPE [12] is composed by the emotion extraction and cause extraction step and the pairing and filtering step. As it is not an end-to-end model, the accumulated error of the first step will affect the result in the second step. End-to-end models [13,14] consider the mutual interaction between emotion and cause. However, the process of mutual interaction is simply implemented by transferring information from one decoder to another uniaxially, thus harming the mutual transfer of information. Moreover, some of the end-to-end models of [14,15] contain a huge number of parameters, resulting in an excessive training time and unremarkable model performance. In this paper, our target is to effectively extract emotion–cause pairs from all potential clause pairs with fewer parameters.

To address the aforementioned challenges, we propose an end-to-end model, Emiece, that predicts the emotion–cause pair from the raw document. We observe that ECPE can be viewed as three mutually related tasks: a primary task of predicting pairings and two auxiliary tasks of predicting emotion and cause clauses respectively. To efficiently learn the three tasks, we consider adopting multi-task learning to establish the connection of them. Multi-task learning [16–28] is an effective way to achieve better generalization performance with a group of related tasks, while sharing some common parameters. Since the two auxiliary tasks are highly similar, we attempt to utilize multi-task learning to tackle the problem of ECPE models’ unsatisfactory performance. Inspired by [29], we choose to leverage a soft-sharing approach to exploit the intrinsic connection between emotion and cause.

After the word-level encoder, each clause is passed through the emotion and cause clause probabilities that indicate the importance of the clause in terms of emotion and cause, and then we obtain the weighted representation of emotion clause and cause clause, respectively. The aim of the weighted representation is to make the true representation and the false one as separate as possible in the feature space so that the extracted representation can be more easily classified for the pair predictor in the higher layer. We conduct sufficient experiments on an ECPE task suitable corpus, which is adapted from the English-language corpus benchmark of the NTCIR-13 Workshop [30].

The main contributions of our work can be summarized as follows:

- **Mutual transfer of information in emotion and cause extraction.** Soft-sharing is applied between emotion and cause encoders. We add the soft-sharing loss to the total loss function in a multi-task learning style to involve mutual interaction between the two auxiliary tasks. Therefore, the two encoders can learn from each other rather than unidirectional learning in previous methods.
- **Efficient pair extractor with weighted representation.** We utilize the weighted representation of emotion and cause to filter the clauses which tend to be meaningless. Therefore, only the useful emotion-weighted and cause-weighted clause representations can be reserved to improve the efficiency of emotion–cause paring.
- **Novel end-to-end ECPE model.** We propose a novel end-to-end method that uses two LSTMs to automatically transfer information between the emotion encoder and cause encoder via soft sharing. Since the end-to-end model considers single emotion and cause extraction along with emotion–cause pairing at the same time, it greatly avoids the cumulative errors in separated steps and significantly improves the performance.

2. Related Work

The emotion–cause extraction (ECE) task was first proposed in [1]. As a word-level task, the extraction is fulfilled with traditional machine learning and rule-based approaches [5–11,31–34]. For example, in [7], the authors proposed a fine-grained rule-based method for the task and conducted experiments on the Chinese microblog posts corpus labeled by human annotators. Despite the overall performance of the word-level task not being promising enough, it provides a new way to look at the emotion classification task.

Another kind of emotion–cause extraction task is based on a clause that solves the problem of word-level labeling in previous work [35–39]. In [35], the authors employed a multi-kernel learning method for the clause-level task on a Chinese emotion cause corpus. Moreover, with the development of deep learning, multiple recurrent neural networks (RNNs) related models have been proposed to solve clause-level tasks due to their excellent performance in analyzing the relationship between different sequences [40,41]. Long short-term memory (LSTM), an advanced version of RNNs, achieves better performance in related tasks thanks to its forgetting mechanism [38]. Although clause-level methods relax word-level annotations into clause-level annotations and achieve higher performance due to the development of neural networks, it is still restricted by manual annotations. In addition, it neglects the mutual relationship and interaction between emotion and cause [42].

To overcome the mentioned drawbacks of ECE, Ref. [12] proposed emotion–cause pair extraction (ECPE) and they conducted a two-step hierarchical structure network for the task. This model separates the emotion and cause extraction and the pairing into two steps; therefore, the mistakes made in the first step will affect the results of the second step.

To solve these limitations in [12], several end-to-end ECPE models have been proposed [13–15,43]. Ref. [14] proposed to model interactions in emotion–cause pairs by means of a two-dimensional transformer [44], which in turn represents emotion–cause pairs in a 2D form. Then a joint framework [45] is used to integrate the two-dimensional representation, interaction and prediction. The work of [15] introduces multi-label learning (MLL) in the ECPE task. The emotion clause and cause clause are designated as the center of the multi-label learning window [46], respectively, and the window slides as the center position is moved. The two joint parts are integrated to obtain the ultimate result. These two models achieved state-of-the-art performance in the ECPE task. Nonetheless, the enormous amount of parameters makes the training overhead of both models extremely large. After that, the method of Ref. [13] uses Bi-LSTM to perform word-level embedding for the input clauses and encodes representations for the emotion and reason clauses.

Finally, a layer of fully connected network is used to predict the matching pairs. It achieves comparable performance with fewer parameters and a simpler architecture. However, its unsatisfactory performance suggests that it does not fully exploit the intrinsic connection between sentiment and cause.

3. Materials and Methods

3.1. Task Formalization

Formally, the documents consist of texts that are segmented as an ordered set of clauses $\mathcal{D} = \{c_1, c_2, \ldots, c_d\}$. The ECPE task aims to figure out a set of emotion–cause pairs

$$\mathcal{P} = \{\ldots, (c_i, c_j), \ldots\}(c_i, c_j \in \mathcal{D}), \tag{1}$$

where c_i is an emotion clause and c_j is the corresponding cause clause. All we have to do is to construct an end-to-end sentiment–cause matching model, which predicts the set of matching pairs $\hat{\mathcal{P}}$, where the correctly predicted part constitutes the set $\hat{\mathcal{P}}_C$, making the $\hat{\mathcal{P}}_C$ as close as possible to the target set $\mathcal{P}$.

3.2. Architecture

The whole model contains three layers as illustrated in Figure 2: word-level encoder layer, clause-level encoder layer, and pairing layer. We take the vector representation $\boldsymbol{v}_{ij}$ of the j-th word in the i-th clause as input. For each clause, the word vector sequence $\boldsymbol{v}_{i,1}, \boldsymbol{v}_{i,2}, \ldots, \boldsymbol{v}_{i,m}$ is passed through a word-level encoder, implemented via a Bi-LSTM with attention [47]. The word-level encoder outputs a clause representation s_i for each clause.

The higher level contains two clause-level encoders implemented for emotion clause detection and cause clause detection, respectively. Any of the current mainstream encoders, such as the stacked Bi-LSTM [48] or BERT [49], can be used. The two encoders take the clause representation sequence $\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_d$ as input and generate the emotion and cause representation of the clauses $\boldsymbol{r}_i^e, \boldsymbol{r}_i^c$. In order to mutually transfer the information obtained by the encoders, we use a soft-sharing strategy between the two encoders. The representations are then fed into detectors (logistic regression layers) to obtain the probability distribution $\boldsymbol{a}_i^e$, $\boldsymbol{a}_i^c$ of the clause being an emotion clause and a cause clause, respectively, formed as

$$\begin{aligned} \boldsymbol{a}_i^e &= \text{softmax}(\boldsymbol{W}^e \boldsymbol{r}_i^e + \boldsymbol{b}^e), \\ \boldsymbol{a}_i^c &= \text{softmax}(\boldsymbol{W}^c \boldsymbol{r}_i^c + \boldsymbol{b}^c), \end{aligned} \tag{2}$$

where $\boldsymbol{W}^e$, $\boldsymbol{W}^c$, $\boldsymbol{b}^e$, $\boldsymbol{b}^c$ are the parameters for emotion and cause detection layers. It should be noted that $\boldsymbol{a}_i^e$ is a 1×2 vector in which one element p_i^e represents the probability that it is indeed an emotion clause and the other element represents the probability that it is not. $\boldsymbol{a}_i^c$ and p_i^c are related as above. It is worth noting that these two outputs $\boldsymbol{a}_i^e$ and $\boldsymbol{a}_i^c$ in our approach also need to participate in pair extraction later, instead of having only one function of obtaining supervisory signals, as in the two auxiliary task outputs in the work of [13]. Thus, in contrast to the cascade of the hierarchical framework of [13], our model achieves parallelism.

These probabilities can be regarded as the attention of emotion and cause encoders. Thus, we multiply the clause representation by the probabilities to obtain the emotion-weighted and cause-weighted clause representation $\tilde{\boldsymbol{r}}_i^e$, $\tilde{\boldsymbol{r}}_i^c$ as

$$\begin{aligned} \tilde{\boldsymbol{r}}_i^e &= p_i^e \boldsymbol{r}_i^e, \\ \tilde{\boldsymbol{r}}_i^c &= p_i^c \boldsymbol{r}_i^c, \end{aligned} \tag{3}$$

Once we collect all the weighted representation of clauses into two sets $\mathcal{E} = \tilde{\boldsymbol{r}}_1^e, \tilde{\boldsymbol{r}}_2^e, \ldots, \tilde{\boldsymbol{r}}_d^e$ and $\mathcal{C} = \tilde{\boldsymbol{r}}_1^c, \tilde{\boldsymbol{r}}_2^c, \ldots, \tilde{\boldsymbol{r}}_d^c$, the Cartesian product is applied on the two sets to generate all the potential emotion–cause pairs $(\tilde{\boldsymbol{r}}_i^e, \tilde{\boldsymbol{r}}_i^c)$. In Figure 3, we use $\boldsymbol{r}_{ij}^p = \tilde{\boldsymbol{r}}_i^e \oplus \tilde{\boldsymbol{r}}_i^c \oplus \boldsymbol{d}_{ij}$ as the representation of a pair, where $\oplus$ denotes the concatenation operator and $\boldsymbol{d}_{ij}$ is the positional embedding vector that indicates the relative position relation between clause i and j [50].

The pairs are fed into the pairing layer one at a time to obtain the predicted label. The pairing layer is a fully connected layer as

$$\begin{aligned} \boldsymbol{h}_{ij} &= \mathrm{ReLU}(\boldsymbol{W}^h \boldsymbol{r}_{ij}^p + \boldsymbol{b}^h), \\ \hat{\boldsymbol{y}}_{ij}^p &= \mathrm{softmax}(\boldsymbol{W}^y \boldsymbol{h}_{ij} + \boldsymbol{b}^y), \end{aligned} \tag{4}$$

where $\hat{\boldsymbol{y}}_{ij}^p$ gives the Bernoulli distribution probabilities of (c_i, c_j) to be an emotion–cause pair. In total, there are three tasks: one primary task for predicting pairs and two auxiliary tasks for predicting emotion and cause clauses. The outputs are $\hat{\boldsymbol{y}}_{ij}^p, \boldsymbol{a}_i^e, \boldsymbol{a}_i^c$, respectively.

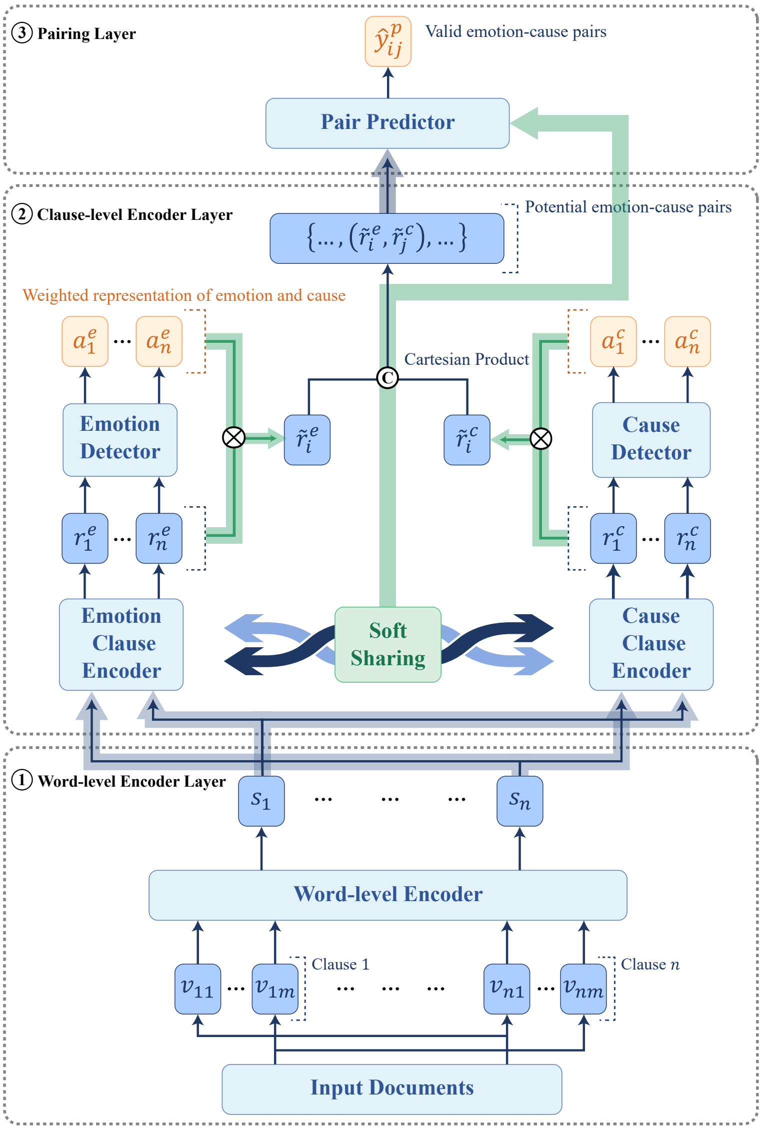


Figure 2. An illustration of our proposed end-to-end mutually interactive emotion–cause pair extractor. v_{11}, v_{1m}, v_{n1} and v_{nm} denote the word vector sequence. $\boldsymbol{s}_1$ and $\boldsymbol{s}_n$ are the clause representation as the output of the word-level encoder. Additionally, $\boldsymbol{r}_1^e, \boldsymbol{r}_n^e, \boldsymbol{r}_1^c$ and $\boldsymbol{r}_n^c$ are the emotion and cause representation of corresponding clauses. $\boldsymbol{a}_1^e, \boldsymbol{a}_n^e, \boldsymbol{a}_1^c$ and $\boldsymbol{a}_n^c$ denote the probability distribution of the clause being an emotion clause and a cause clause. $\tilde{\boldsymbol{r}}_i^e$ and $\tilde{\boldsymbol{r}}_i^c$ show the emotion-weighted and cause-weighted clause representations, respectively. $(\tilde{\boldsymbol{r}}_i^e, \tilde{\boldsymbol{r}}_j^c)$ represents potential emotion–cause pairs. $\hat{\boldsymbol{y}}_{ij}^p$ gives the Bernoulli distribution probabilities of potential emotion–cause pairs to be true.

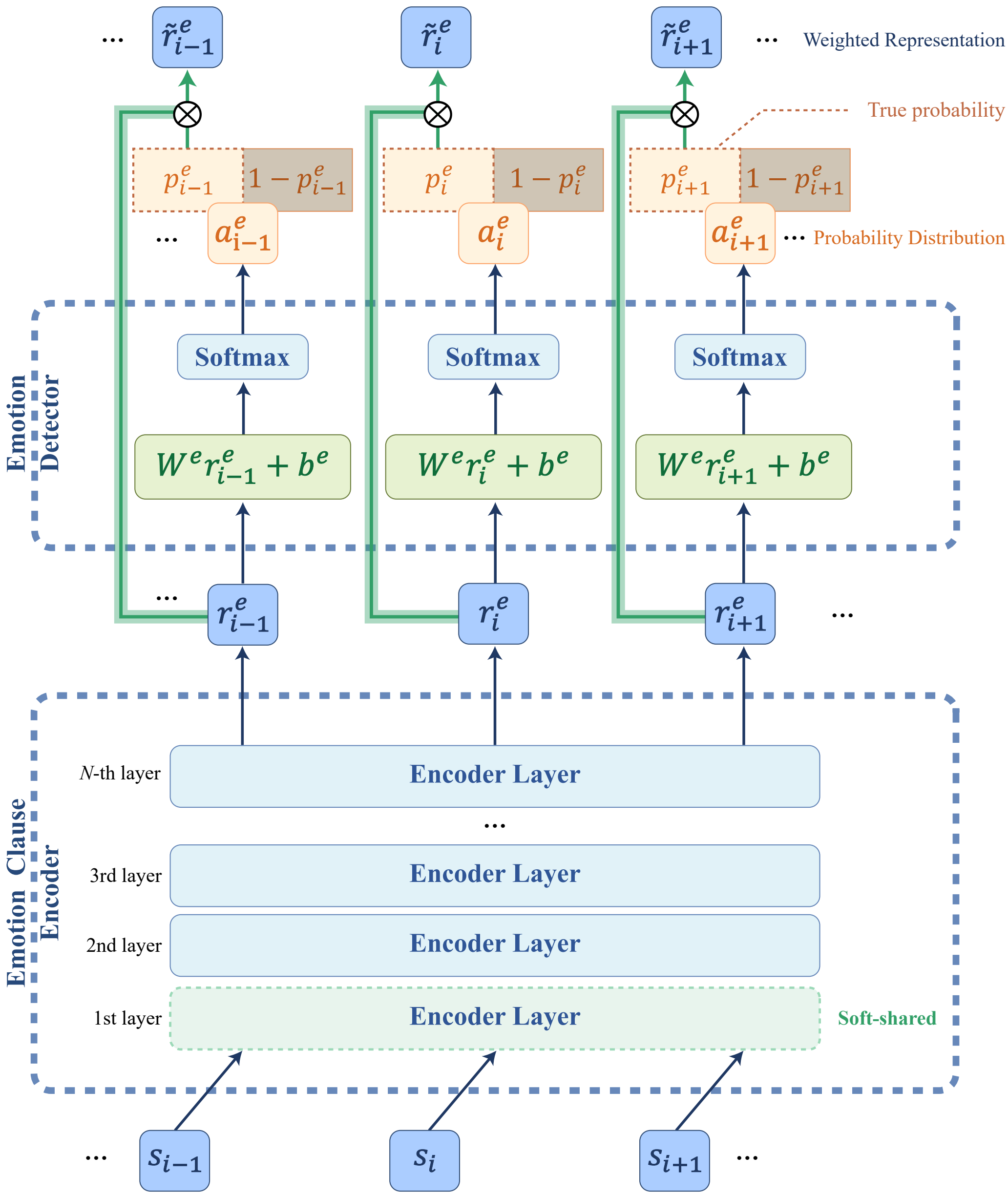


Figure 3. Implementation details of the weighted representation. The weighted representation of the cause clause is similar to that of the emotion clause, thus it is omitted here. s_{i-1}, s_i and s_{i+1} are the clause representation output by the word-level encoder. r^e_{i-1}, r^e_i and r^e_{i+1} are the emotion and cause representations of corresponding clauses.

3.3. Learning with Mutual Transfer of Information

In order to effectively train Emiece, we set the loss function to be

$$L_{\text{total}} = \lambda_p L_p + \lambda_e L_e + \lambda_c L_c + \lambda_{sf} L_{sf}, \tag{5}$$

where L_p, L_e and L_c are the cross-entropy losses of pair extraction, emotion clause detection, and cause clause detection, respectively. L_{sf} is the soft-sharing loss for the mutual transfer of information.

Following [29], we define

$$L_{sf} = \sum_{d \in \mathcal{D}} \|\phi^e_d - \phi^c_d\|^2, \tag{6}$$

where $\mathcal{D}$ is the set of sharing parameter indices, ϕ^e and ϕ^c are the emotion and cause encoder parameters, respectively. The works in recent years found [51] that features extracted in the shallow layers of a deep neural network contain more general features of different tasks. The higher the network level, the more task specific the extracted features will be. Following the work of [29,52], we employ a similar soft-sharing strategy on the first-layer encoder in the emotion and cause encoders and keep the second layer of the two encoders unshared, which is further away from the input. A discussion of soft-sharing modules and ablation studies is given in Section 5.2.

To avoid the imbalance of positive pairs and negative pairs in the pair extraction, the loss L_p is revised as

$$L_p = L_p^+ + \lambda_- L_p^-, \tag{7}$$

where L_p^+,L_p^- denotes the term of positive and negative ground truths in the cross-entropy loss function. λ_- is relatively small since the number of negative pairs is much more than positive ones.

3.4. Evaluation Metrics

Following the previous work [12], we also used the same three evaluation metrics: precision, recall, and F1 score. The F1 score takes both precision and recall into account, and thus it is the most crucial of these evaluation metrics. They are defined as follows:

$$\begin{aligned} \text{Precision} &= \frac{|\hat{\mathcal{P}}_c|}{|\hat{\mathcal{P}}|}, \\ \text{Recall} &= \frac{|\hat{\mathcal{P}}_c|}{|\mathcal{P}|}, \\ \text{F1 score} &= \frac{2 \times Precision \times Recall}{Precision + Recall}, \end{aligned} \tag{8}$$

according to Section 3.1, $|\hat{\mathcal{P}}_c|$ indicates the number of emotion–cause pairs predicted by the model, $|\hat{\mathcal{P}}|$ indicates the number of correct pairs among these predicted matches, and $|\mathcal{P}|$ indicates the number of all emotion–cause pairs in the actual dataset.

4. Results

4.1. Dataset

The dataset was constructed by [13] from an existing emotion–cause extraction (ECE) corpus. The corpus was introduced in the NTCIR-13 Workshop [30] for the ECE challenge. There are 2843 documents taken from several English novels, and each document is annotated with the following:

1. Emotion–cause pairs (the set of emotion clauses and their corresponding cause clauses);
2. Emotion category of each clause;
3. Keywords in the emotion clauses.

Detailed statistics about the dataset are presented in Tables 1 and 2 below. In our experiments, we do not leverage the emotion category or keywords and only exploit the emotion–cause pairs in the training process. None of the annotations are used when testing the model. The whole dataset is split by 80%–10%–10% for training, validating, and testing.

Table 1. Overall information on the English-language corpus dataset. Information on the four main dimensions related to the ECPE task was counted.

# Document	# Clause ($\lvert\mathcal{D}\rvert$)	# Emotion-Cause Pair ($\lvert\mathcal{P}\rvert$)	# Annotated Emotion Type
2843	21,802	3272	6

Table 2. The distribution of emotion clauses corresponding to the six annotated emotions in the dataset.

Annotated Emotion	# Corresponding Emotion Clause
happiness	741
surprise	388
sadness	638
fear	622
anger	269
disgust	214

4.2. Baselines and Settings

We include four baseline methods in our comparison with `Emiece`: the original two-step model `ECPE` [12], a relatively lightweight end-to-end model `E2E-PExt`$_{\mathtt{E}}$ [13], and two state-of-the-art ECPE models `ECPE-2D(BERT)` [14] and `ECPE-MLL(ISML-6)` [15].

- `ECPE` [12]: As a second step, a Cartesian product is applied to the emotion clauses and causal relationships extracted from the multi-task learning network in the first step in order to compose them into pairs, and a filtering model is trained so that the pair containing the causal relationship is the final output. Bi-LSTM and attention [47,48] is the word-level encoder used in the first extraction step, and Bi-LSTM [48] is used in the emotion and cause extractors as well. Logic regression is used to filter the pairs in the second step.
- `ECPE-2D(BERT)` [14]: The interactions in the emotion–cause pairs were modeled by a 2D transformer, which in turn represented the pairs in a two-dimensional form, i.e., a square matrix. Two-dimensional representations are integrated with interactions and predictions using a joint framework. The encoding part uses a word-level Bi-LSTM and an attention mechanism [47], while the clause-level emotion extractor and cause extractor leverage BERT [49] to enhance the overall effectiveness of the model.
- `ECPE-MLL(ISML-6)` [15]: Multi-label learning (MLL) was introduced in the ECPE task. To obtain a representation of the clause, the emotion clause and cause clause are specified as the center of the multi-label learning window. An iterative synchronous multi-task learning (ISML) model with six iterations is used for clause encoding, while the same Bi-LSTM [47] is used for word-level embedding.
- `E2E-PExt`$_{\mathtt{E}}$ [13]: Using Bi-LSTM plus attention [47], the clause-level representation is obtained based on the word-level one. The clause level representation uses another Bi-LSTM network to further extract contextual information and is used to determine whether the clause is an emotion one or a cause one. Finally, the predicted pair is obtained by a fully connected neural network.

We denote Emiece-LSTM that using as the stacked Bi-LSTM [48] clause-level encoder. Emiece-LSTM is trained for 30 epochs using the Adam optimizer [53]. We set the learning rate α = 0.005, and batch size N = 64. The model parameters ϕ are initialized randomly following uniform distribution $\phi \sim U(-0.1, 0.1)$. We leverage GloVe word embedding [54] of 200 dimensions. The dropout rate is set to 0.8 for word embeddings and ℓ_2 decay is set to 10^{-5} on softmax parameters. The loss weights are set as λ_e:λ_c:λ_p:λ_{sf} = 1:1:2.5:0.75 and the negative pair weight λ_- = 0.4. Here, the value of λ_- is taken with reference to the work of [13], and we have made many attempts to take the value of λ_{sf}, as shown in Figure 4. As a result of setting λ_{sf} to 0.75, the model performs best in combination at validation, i.e., it performs best on the more important F1-score metric and relatively well on the other two metrics. We also denote Emiece-BERT with the clause-level encoder setting to BERT [49]. Constrained training server performance, we set the batch size of Emiece-BERT to 16 and vary the learning rate by 2×10^{-5}, keeping other hyperparameters the same.

In order to achieve higher model performance through better positional embeddings, randomly initialized embeddings are trained after setting the clipping distance [50] to 10.

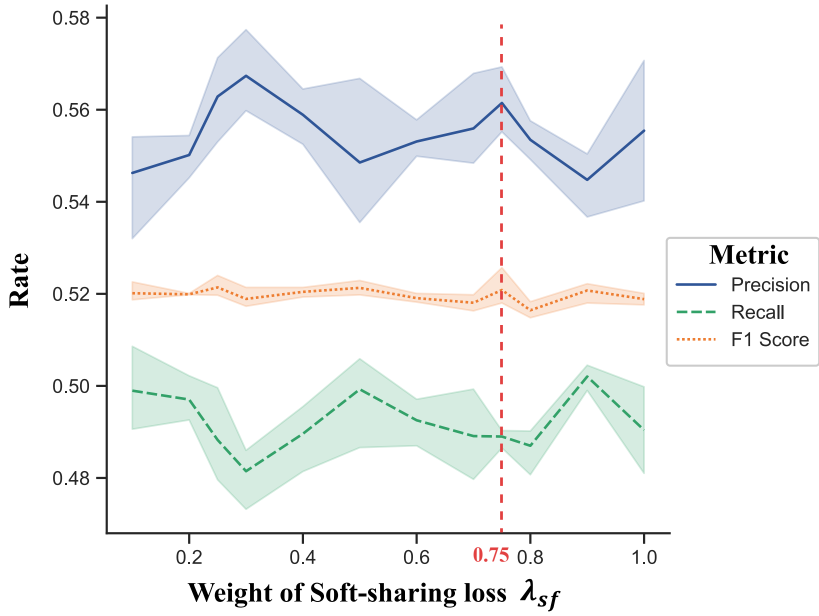


Figure 4. Repeated experiments on the determination of the value of λ_{sf} in different metric. A higher percentage value and a smaller variance indicate better results. It is not appropriate to use weights that are too high or too low.

4.3. Overall Performance

Table 3 presents the experimental performance in the ECPE task. Compared to a similar end-to-end approach E2E-PExt$_E$ [13], the F1 scores of our model Emiece-LSTM improved by 1.28%, 1.72% and 2.4% in the emotion extraction, cause extraction and emotion–cause pair extraction tasks, respectively. This is a strong indication that the way of using the intrinsic connection between emotion and cause to extract match pairs is correct, and the related ablation experiments will be placed in Section 5. Compared to the traditional two-step approach ECPE [12], our Emiece model improves the performance of the emotion–cause pair extraction task by an impressive 8.9%. In addition, our method has better results than the current methods ECPE-2D(BERT) [14] and ECPE-MLL(ISML-6) [15] on the most dominant prediction matching pair task, even though the number of parameters is much smaller than theirs.

Table 3. Best results of our model and previous experimental methods with existing metrics after hyper-parameter tuning. Bold fonts indicate the best results for the method, and underlining stands for the second best results. The top half is the performance of pipelines, and the bottom half is the one of our model.

	Emotion Extraction			Cause Extraction			Pair Extraction		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score
`ECPE` [12]	0.6741	<u>0.7160</u>	0.6940	0.6039	0.4734	0.5301	0.4694	0.4102	0.4367
`ECPE-2D(BERT)` [14]	0.7435	0.6968	0.7189	0.6491	0.5353	0.5855	<u>0.6049</u>	0.4384	0.5073
`ECPE-MLL(ISML-6)` [15]	0.7546	0.6996	<u>0.7255</u>	0.6350	**0.5919**	<u>0.6110</u>	0.5926	0.4530	0.5121
`E2E-PExt`$_E$ [13]	0.7163	0.6749	0.6943	0.6636	0.4375	0.5226	0.5134	<u>0.4929</u>	0.5017
`Emiece-LSTM` (Ours)	<u>0.7702</u>	0.6550	0.7071	<u>0.7010</u>	0.4413	0.5398	0.5693	0.4903	<u>0.5257</u>
`Emiece-BERT` (Ours)	**0.8263**	**0.7441**	**0.7830**	**0.7135**	<u>0.5522</u>	**0.6225**	**0.6833**	**0.5325**	**0.5985**

Through the case study in Figure 5, we can also visualize that the Emiece method performs well in relatively complex short texts.

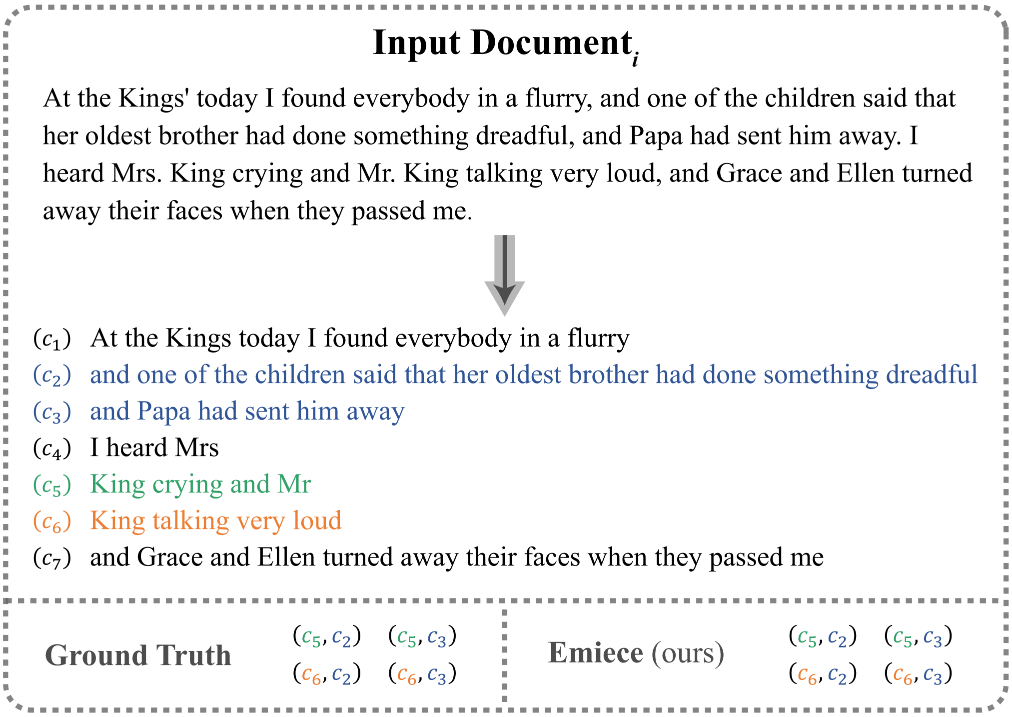


Figure 5. A case study of our method. Our method also works well in extracting emotion–cause pairs when the input text contains multiple emotions and multiple causes that match each other.

5. Ablation Study

We studied the effect of different modules on the experimental results through ablation experiments.

5.1. Clause-Level Encoder

The excellent performance of ECPE-2D(BERT) [14] and ECPE-MLL(ISML-6) [15] with a larger number of parameters in the ECPE task cannot be ignored. Inspired by their work, we replaced the clause-level encoder in Emiece from the stacked Bi-LSTM [48] to BERT [49] as the new Emiece-BERT model. Nonetheless, due to the server storage space limitation, training Emiece-BERT can only set the batch size to 16. To control the variables, we keep the hyperparameters of Emiece-LSTM in line with Emiece-BERT and retrain it.

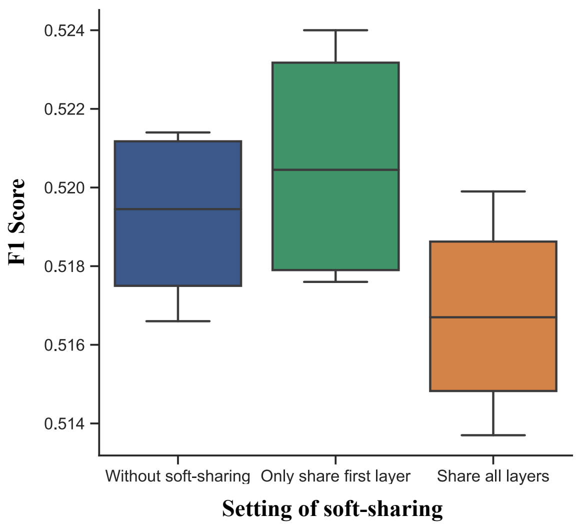
As can be seen from Table 4, under the end-to-end soft-sharing emotion–cause pair extraction framework, simply replacing a more complex encoder, i.e., BERT [49], makes the model surprisingly effective in improving each metric on each of the three tasks of ECPE. However, this undoubtedly introduces a much larger number of parameters for training. In total, 69 hours are consumed by the BERT model in one training, while 16 hours are consumed by Emiece-LSTM training; such a replacement causes a considerable overhead and makes the model redundant and unwieldy.

Table 4. Comparison of the model effects of Emiece composed of two different clause-level encoders. The results are under the same hyper-parameter setting.

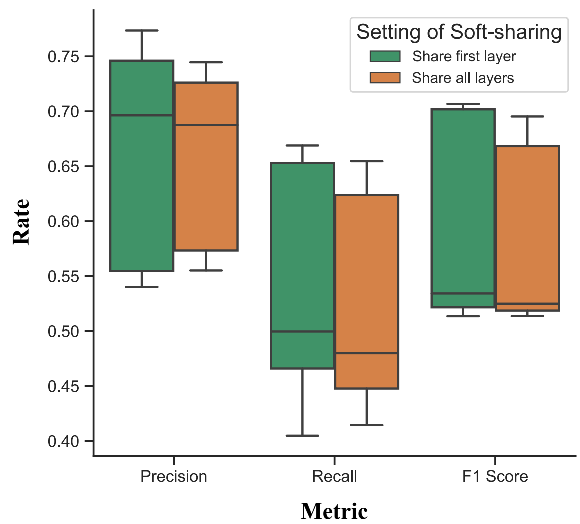
Clause-Level Encoder	Emotion Extraction			Cause Extraction			Pair Extraction		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score
`stacked Bi-LSTM` [47]	0.7425	0.6665	0.7014	0.6570	0.4762	0.5473	0.5398	0.4973	0.5153
`BERT` [49]	**0.8263**	**0.7441**	**0.7830**	**0.7135**	**0.5522**	**0.6225**	**0.6833**	**0.5325**	**0.5985**

5.2. Mutual Transfer of Information

We selected the most comprehensive metric F1 score and the primary task emotion–cause pair extraction to analyze the effectiveness of the mutual transfer of information by varying the soft-sharing setting: no soft-sharing, sharing only the first layer of the emotion and cause encoders, and sharing all layers of both encoders. Figure 6a shows that sharing only the first layer of the encoder achieves better performance compared to the other two cases, indicating that the intrinsic connection between emotion and cause is closer to the lexico-syntactic level [29]. Due to the long training time of model Emiece-BERT, only model Emiece-LSTM is selected here.



(a) F1 score for different soft-sharing settings in ECPE tasks.



(b) Performance of two soft-sharing settings with different metrics.

Figure 6. (**a**) F1 score for different soft-sharing settings in emotion–cause pair extraction tasks. Soft sharing of first layer parameters is far better than not sharing. (**b**) A detailed comparison of the performance of soft sharing one layer and sharing all layers under different metrics. More layers of soft-sharing parameters do not directly lead to better results.

In Figure 6b, the results illustrate that the number of parameters shared in the encoder is not linearly correlated with the effect of the prediction results. This is consistent with the results of previous experiments in which the weight of soft sharing was altered by changing the value of λ_{sf}, but the higher the weight is not better. Ref. [52] found that in the seq2seq machine translation model, the low-level layer of the RNNs unit (i.e., the first layer in the encoder) represents the word structure, while the high-level layer focuses on the semantic meaning. Since the semantic information of the emotion clause and the cause clause is quite different, sharing the high-level layer alone will blur the features of the text, not to mention that sharing all layers will confuse the word structure information with the semantic information, resulting in a reduced effect.

6. Conclusions

In this paper, we propose an end-to-end model that mutually transfers information via soft sharing between emotion and cause extraction tasks. By using weighted representations of sentiment and cause filtering nonsensical clauses, we improved the efficiency of emotion–cause pairing. The end-to-end model takes into account both emotion and cause extraction and emotion–cause pairing, thereby greatly reducing the cumulative error. Based on experiments conducted on the standard ECPE dataset, Emiece achieves significant improvements in emotion–cause pairs extraction over the original two-step ECPE model and other end-to-end models.

In the future, (1) we will examine the reasons for the model's mistakes by performing an interpretability analysis on the possible wrong predictions in the current model. (2) We also attempt to reduce the network depth in order to perform the emotion–cause pair extraction as a solution to the problem that it is difficult to train a clause-level encoder with numerous parameters. (3) It is relatively complex to construct pairs with the Cartesian product, so we will use a more efficient module for pairing prediction.

Author Contributions: Conceptualization, H.X.; Software, Z.L.; Writing—original draft, T.M.; Writing—review & editing, B.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Key R&D Program of China [Grant No. 2018AAA0100500]; National Natural Science Foundation of China [Grant No. 61906040]; the Natural Science Foundation of Jiangsu Province [Grant Numbers BK20190335, BK20190345]; National Natural Science Foundation of China [Grant Numbers 61906037, 61972085]; the Fundamental Research Funds for the Central Universities; Jiangsu Provincial Key Laboratory of Network and Information Security [Grant No. BM2003201], Key Laboratory of Computer Network and Information Integration of Ministry of Education of China [Grant No. 93K-9].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lee, S.Y.M.; Chen, Y.; Huang, C.R. 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, Los Angeles, CA, USA, 5 June 2010; pp. 45–53.
2. Russo, I.; Caselli, T.; Rubino, F.; Boldrini, E.; Martínez-Barco, P. EMOCause: An Easy-adaptable Approach to Extract Emotion Cause Contexts. In Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA 2.011), Portland, OR, USA, 24 June 2011; pp. 153–160.
3. Gui, L.; Xu, R.; Wu, D.; Lu, Q.; Zhou, Y. Event-driven emotion cause extraction with corpus construction. In *Social Media Content Analysis: Natural Language Processing and Beyond*; World Scientific: Singapore, 2018; pp. 145–160.
4. Gui, L.; Xu, R.; Lu, Q.; Wu, D.; Zhou, Y. Emotion cause extraction, a challenging task with corpus construction. In Proceedings of the Chinese National Conference on Social Media Processing, Beijing, China, 1–2 November 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 98–109.
5. Chen, Y.; Lee, S.Y.M.; Li, S.; Huang, C.R. Emotion cause detection with linguistic constructions. In Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), Beijing, China, 23–27 August 2010; pp. 179–187.
6. Gao, K.; Xu, H.; Wang, J. Emotion cause detection for chinese micro-blogs based on ecocc model. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Ho Chi Minh City, Vietnam, 19–22 May 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 3–14.
7. Li, W.; Xu, H. Text-based emotion classification using emotion cause extraction. *Expert Syst. Appl.* **2014**, *41*, 1742–1749. [CrossRef]
8. Li, X.; Song, K.; Feng, S.; Wang, D.; Zhang, Y. 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, 31 October–4 November 2018; Association for Computational Linguistics: Brussels, Belgium, 2018; pp. 4752–4757.
9. Xu, B.; Lin, H.; Lin, Y.; Diao, Y.; Yang, L.; Xu, K. Extracting emotion causes using learning to rank methods from an information retrieval perspective. *IEEE Access* **2019**, *7*, 15573–15583. [CrossRef]
10. Yu, X.; Rong, W.; Zhang, Z.; Ouyang, Y.; Xiong, Z. Multiple Level Hierarchical Network-Based Clause Selection for Emotion Cause Extraction. *IEEE Access* **2019**, *7*, 9071–9079. [CrossRef]
11. Neviarouskaya, A.; Aono, M. Extracting causes of emotions from text. In Proceedings of the Sixth International Joint Conference on Natural Language Processing, Nagoya, Japan, 14–18 October 2013; pp. 932–936.
12. Xia, R.; Ding, Z. Emotion-Cause Pair Extraction: A New Task to Emotion Analysis in Texts. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 1003–1012.
13. Singh, A.; Hingane, S.; Wani, S.; Modi, A. An End-to-End Network for Emotion-Cause Pair Extraction. In Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Online, 19 April 2021; pp. 84–91.
14. Ding, Z.; Xia, R.; Yu, J. ECPE-2D: Emotion-cause pair extraction based on joint two-dimensional representation, interaction and prediction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 3161–3170.

15. Ding, Z.; Xia, R.; Yu, J. End-to-end emotion-cause pair extraction based on sliding window multi-label learning. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 3574–3583.
16. Caruana, R. Multitask learning. *Mach. Learn.* **1997**, *28*, 41–75. [CrossRef]
17. Ben-David, S.; Schuller, R. Exploiting task relatedness for multiple task learning. In *Learning Theory and Kernel Machines*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 567–580.
18. Evgeniou, T.; Micchelli, C.A.; Pontil, M.; Shawe-Taylor, J. Learning multiple tasks with kernel methods. *J. Mach. Learn. Res.* **2005**, *6*, 615–637.
19. Argyriou, A.; Evgeniou, T.; Pontil, M. Multi-task feature learning. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 4–7 December 2006; Volume 19.
20. Kumar, A.; Daume, H., III. Learning task grouping and overlap in multi-task learning. *arXiv* **2012**, arXiv:1206.6417.
21. Luong, M.T.; Le, Q.V.; Sutskever, I.; Vinyals, O.; Kaiser, L. Multi-task sequence to sequence learning. *arXiv* **2015**, arXiv:1511.06114.
22. Misra, I.; Shrivastava, A.; Gupta, A.; Hebert, M. Cross-stitch networks for multi-task learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3994–4003.
23. Hashimoto, K.; Xiong, C.; Tsuruoka, Y.; Socher, R. A Joint Many-Task Model: Growing a Neural Network for Multiple NLP Tasks. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 9–11 September 2017; pp. 1923–1933.
24. Pasunuru, R.; Bansal, M. Multi-Task Video Captioning with Video and Entailment Generation. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; Association for Computational Linguistics: Stroudsburg, PA, USA, 2017; pp. 1273–1283. [CrossRef]
25. Tan, X.; Ma, T.; Su, T. Fast and Privacy-Preserving Federated Joint Estimator of Multi-sUGMs. *IEEE Access* **2021**, *9*, 104079–104092. [CrossRef]
26. Xu, H.; Wang, M.; Wang, B. A Difference Standardization Method for Mutual Transfer Learning. In Proceedings of the 39th International Conference on Machine Learning, Baltimore, MD, USA, 17–23 July 2022; Volume 162, pp. 24683–24697.
27. Zhang, J.; Yan, K.; Mo, Y. Multi-Task Learning for Sentiment Analysis with Hard-Sharing and Task Recognition Mechanisms. *Information* **2021**, *12*, 207. [CrossRef]
28. Fan, C.; Yuan, C.; Gui, L.; Zhang, Y.; Xu, R. Multi-task sequence tagging for emotion-cause pair extraction via tag distribution refinement. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 2339–2350. [CrossRef]
29. Guo, H.; Pasunuru, R.; Bansal, M. Soft Layer-Specific Multi-Task Summarization with Entailment and Question Generation. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; Association for Computational Linguistics: Stroudsburg, PA, USA, 2017; pp. 1273–1283.
30. Gao, Q.; Hu, J.; Xu, R.; Gui, L.; He, Y.; Wong, K.F.; Lu, Q. Overview of NTCIR-13 ECA Task. In Proceedings of the NTCIR-13 Conference, Tokyo, Japan, 5–8 December 2017.
31. Gao, K.; Xu, H.; Wang, J. A rule-based approach to emotion cause detection for Chinese micro-blogs. *Expert Syst. Appl.* **2015**, *42*, 4517–4528. [CrossRef]
32. Yada, S.; Ikeda, K.; Hoashi, K.; Kageura, K. A bootstrap method for automatic rule acquisition on emotion cause extraction. In Proceedings of the 2017 IEEE International Conference on Data Mining Workshops (ICDMW), New Orleans, LA, USA, 18–21 November 2017; pp. 414–421.
33. Ghazi, D.; Inkpen, D.; Szpakowicz, S. Detecting emotion stimuli in emotion-bearing sentences. In Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics, Cairo, Egypt, 14–20 April 2015; pp. 152–165.
34. Cheng, X.; Chen, Y.; Cheng, B.; Li, S.; Zhou, G. An emotion cause corpus for chinese microblogs with multiple-user structures. In *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*; Association for Computing Machinery: New York, NY, USA, 2017; Volume 17, pp. 1–19.
35. Serban, I.V.; Sordoni, A.; Lowe, R.; Charlin, L.; Pineau, J.; Courville, A.; Bengio, Y. A hierarchical latent variable encoder-decoder model for generating dialogues. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 3295–3301.
36. Tang, H.; Ji, D.; Zhou, Q. Joint multi-level attentional model for emotion detection and emotion-cause pair extraction. *Neurocomputing* **2020**, *409*, 329–340. [CrossRef]
37. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; Volume 27.
38. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [CrossRef]
39. Fan, W.; Zhu, Y.; Wei, Z.; Yang, T.; Ip, W.; Zhang, Y. Order-guided deep neural network for emotion-cause pair prediction. *Appl. Soft Comput.* **2021**, *112*, 107818. [CrossRef]
40. Jia, X.; Chen, X.; Wan, Q.; Liu, J. A Novel Interactive Recurrent Attention Network for Emotion-Cause Pair Extraction. In Proceedings of the 2020 3rd International Conference on Algorithms, Computing and Artificial Intelligence, Sanya, China, 24–26 December 2020; pp. 1–9.
41. Chen, F.; Shi, Z.; Yang, Z.; Huang, Y. Recurrent synchronization network for emotion-cause pair extraction. *Knowl.-Based Syst.* **2022**, *238*, 107965. [CrossRef]

42. Yu, J.; Liu, W.; He, Y.; Zhang, C. A mutually auxiliary multitask model with self-distillation for emotion-cause pair extraction. *IEEE Access* **2021**, *9*, 26811–26821. [CrossRef]
43. Li, C.; Hu, J.; Li, T.; Du, S.; Teng, F. An effective multi-task learning model for end-to-end emotion-cause pair extraction. *Appl. Intell.* **2022**, 1–11. [CrossRef]
44. Fan, C.; Yuan, C.; Du, J.; Gui, L.; Yang, M.; Xu, R. Transition-based directed graph construction for emotion-cause pair extraction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 3707–3717.
45. Wei, P.; Zhao, J.; Mao, W. Effective inter-clause modeling for end-to-end emotion-cause pair extraction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 3171–3181.
46. Yang, X.; Yang, Y. Emotion-Type-Based Global Attention Neural Network for Emotion-Cause Pair Extraction. In Proceedings of the International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, Guiyang, China, 24–26 July 2021; pp. 546–557.
47. Bahdanau, D.; Cho, K.H.; Bengio, Y. Neural machine translation by jointly learning to align and translate. In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, USA, 7–9 May 2015.
48. Graves, A.; Mohamed, A.-r.; Hinton, G. Speech recognition with deep recurrent neural networks. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, USA, 26–31 May 2013; pp. 6645–6649.
49. Kenton, J.D.M.W.C.; Toutanova, L.K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the NAACL-HLT, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
50. Shaw, P.; Uszkoreit, J.; Vaswani, A. Self-Attention with Relative Position Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), New Orleans, LA, USA, 1–6 June 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 464–468.
51. Howard, J.; Ruder, S. Universal Language Model Fine-tuning for Text Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 328–339.
52. Belinkov, Y.; Durrani, N.; Dalvi, F.; Sajjad, H.; Glass, J.R. What do Neural Machine Translation Models Learn about Morphology? In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2017; pp. 861–872.
53. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the ICLR (Poster), San Diego, CA, USA, 7–9 May 2015.
54. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.