ABSTRACT

Previous studies about event-level sentiment analysis (SA) usually model the event as a topic, a category or target terms, while the structured arguments (e.g., subject, object, time and location) that have potential effects on the sentiment are not well studied. In this paper, we redefine the task as structured event-level SA and propose an End-to-End Event-level Sentiment Analysis (E³SA) approach to solve this issue. Specifically, we explicitly extract and model the event structure information for enhancing event-level SA. Extensive experiments demonstrate the great advantages of our proposed approach over the state-of-the-art methods. Noting the lack of the dataset, we also release a large-scale real-world dataset with event arguments and sentiment labelling for promoting more researches.

CCS CONCEPTS

- Computing methodologies → Natural language processing;
- Information systems → Web applications.

KEYWORDS

event-level sentiment analysis, structured, datasets

1 INTRODUCTION

Sentiment analysis (SA) has received much attention in both academia and industry for its great value [38, 40]. Instead of assuming that the entire text has an overall sentiment polarity, more and more researchers turn to investigate the fine-grained SA, such as event-level SA [22–24]. Event-level SA aims to identify the feelings or opinions expressed by users on a social platform about real-time events from financial news, sports, weather, entertainment, etc. It is vital for many applications, such as stock prediction [19], public opinion analysis [23].

Previous studies about event-level SA mainly utilize the related snippet or context of the event for SA [9, 10, 19, 22, 41]. Patil and Chavan [22], Petrescu et al. [23] detected bursty topics as the events
via LDA or clustering algorithm and inferred their sentiments. Moreover, Deng and Wiebe [3] recognized sentiments toward entities and events that are several terms in the text. Whereas, these studies focused on the event related text modeling, which neglects the influence of the event’s inherent structure.

According to our previous observations, the event structured arguments, such as subject, object, time and location, play an important role in event-level SA. As shown in Figure 1, the events with the same trigger “increase” indicate opposite sentiments for different subjects, namely “operating costs” and “revenue”. Additionally, for the two events (the second and fourth row in the table) with the same negative sentiment, the objects as “10%-20%” and “64.65%” help reveal the strength of sentiment polarity. Therefore, this work aims to enhance event-level SA with the structured arguments.

Noting that there are few studies about event SA with the fine-grained structured arguments, we reformalize a structured event-level SA task, which extracts the events with arguments and predicts the sentiments. This task suffers from four challenges as follows: (C1) The multi-subtasks (e.g., trigger extraction, argument extraction, sentiment classification) are related with each other, and performing them independently will lead to error propagation; (C2) One document may contain multiple events with different sentiments. Taking Figure 1 as an example, there are four events, the sentiment polarities of them are different among each other; (C3) Unlike general aspect-level SA, the event consists of triggers and arguments associated with their roles, which is harder to model than a topic. Thus, the existing aspect-based SA models can not be applied to this task directly; (C4) Lack of the labeled datasets for this task. The existing datasets mainly focus on event extraction or aspect’s SA, while the sentiment of the structured event is not well studied.

To deal with the above challenges, we present an end-to-end approach for structured event-level SA, which reduces the error propagation among the subtasks (C1). Particularly, we first design a feature-enhanced trigger extractor to extract multiple events’ triggers simultaneously (C2). Second, to better model the event information, we design a trigger-enhanced argument extractor and event-level sentiment classifier, which take trigger and argument information into account (C3). Finally, we collect and label a real-world dataset in the finance domain for this task (C4). This dataset provides a new perspective for SA and a new benchmark for event-level SA.

The main contributions of this paper are summarized as follows.

- We reformalize a structured event-level SA task, which focuses on enhancing the event-level SA with structured arguments.
- To mitigate the effect of error propagation, we propose E3SA to model the relationships among the multiple tasks and multiple events by taking structured elements into account.
- For the lack of the off-the-shelf datasets, we release one large-scale dataset for this task. Also, extensive experiments also show that our model outperforms all the strong baselines significantly.

2 RELATED WORK

In this section, we mainly review the most related papers about event extraction, sentiment analysis and sentiment analysis on events.

Event Extraction. Event extraction (EE) is a critical task in public opinion monitoring and financial field [8, 13, 16, 33]. It mainly has two key methods, pipeline and joint model. The pipeline model, which performs trigger extraction and argument role assignment independently [2, 21], ignoring the relationships among the elements in events. The joint model generally extracts the related elements at the same time [14, 20, 28, 35]. Recently, researchers have paid attention to document-level EE, which is more complex. Yang et al. [31] proposed a DCFE model to extract the event by multiple sentences based on distant supervision. Liu et al. [18] and Zheng et al. [36] proposed to extract multiple events jointly. Du and Cardie [5] regarded EE as a question answering task to extract the event arguments without entity propagation. Unlike the current work, we focus on modeling the events’ sentiment information.

Sentiment Analysis. Sentiment analysis can be divided into three types: sentence-level, document-level and aspect-level SA [1, 17]. We mainly review the most related works about aspect-based SA (ABSA) [24, 37]. This task aims to predict the sentiments of the aspects in the document, where the aspects are categories, topics or target terms. To take the aspect into account, attention-based models were designed to capture the relationships between the aspect and its context [7, 34]. Moreover, position information and syntax information was integrated to better model the aspect, such as a category or target [15, 39]. Applying these models to our task will reduce the performance since we focus on modeling the structured events with complex argument information.

Sentiment Analysis on Events. There are some works namely event-based sentiment analysis [6, 9, 10, 19, 22, 23, 41], which are different from our task. Most of these studies focus on detecting the event via topic model and judging the sentiment of the event, which is a category, topic, or term, while the detailed information (e.g., arguments) is ignored by them. Additionally, they only consider one event in a sentence or document. In fact, a text may consist of multiple events and an event is not only a topic. To address these problems, we propose an end-to-end approach for event-level sentiment analysis, which aims to identify the events and their sentiments.

3 DATASET

Data Collection and Annotation. Due to lack of annotated resources, we collect and annotate a financial corpus with events and their sentiment polarities, and obtain an event-level SA corpus. Specifically, we collect 3500 financial news from the portal website\(^2\). We filter the documents that contain less than 50 words or more than 500 words and finally obtain 3500 short documents. Then, we give annotation guidelines and ten examples to eight human annotators, who manually annotate triggers, arguments and the sentiment labels via Baidu’s EasyData platform\(^3\). Note that we only consider the important events that may influence companies’ stock or users’ decisions. Because there are too many events in the text and some of the events are not useless for the downstream tasks. To ensure the labeling quality, each document is annotated by three

\(^2\)Note that all the texts are open accessed news on the https://www.eastmoney.com/ without personal information.

\(^3\)http://ai.baidu.com/easydata/
annotators in order. Moreover, we randomly select 100 examples and ask another three annotators to label these documents. We measure pairwise inter-annotator agreement on tuples among two versions using Krippendorff’s alpha coefficient [11].

**Data Analysis.** We obtain 3142 samples after filtering the examples without events (Table 1). This dataset has several characteristics: 1) One document always contains multiple events; 2) The events in a document may have different sentiment polarities (#Multi); 3) The arguments contained in the same event may have different sentiment polarities (#Multi-Across). We compare our dataset with the existing ones to clarify the differences. First, EE focuses on extracting the events while ignoring their emotions, and ABSA focuses on the aspects’ sentiments while ignoring the structure information. Our task aims to judge the events’ sentiments, which is more challenging than these two tasks. Second, though most datasets are at the document level, one event or aspect is always in a sentence. In our dataset, one event may be across in multiple sentences. Third, our dataset is larger than or comparable with most datasets. ACE2005 and MUC-4 contain less than 1,500 documents. The size of DOC2EDAG is 32,040, while it is labeled using distant supervision.

## 4 OUR APPROACH

In this paper, we propose a E³SA framework for structured event-level SA (Figure 2). To reduce the propagated errors in the pipeline, we propose a joint approach. E³SA consists of four parts: (i) contextualized word embedding module that models the document with contextual representation; (ii) feature-enhanced trigger extractor that extracts all triggers in the documents with additional features such as POS and NER labels; (iii) trigger-enhanced argument extractor that extracts the arguments concerning the given trigger by taking trigger information into account; (iv) event-level sentiment classifier that judges the events’ sentiment polarities with argument information.

Formally, given a document $d = \{s_1, \ldots, s_{|d|}\}$, where $|d|$ is the number of the sentences, $s_i$ is the $i$-th sentence in the document $d$, which contains $|s_i|$ words, $\{w_1^{(i)}, \ldots, w_{|s_i|}^{(i)}\}$. The goal of this task is to extract all the events $E = \{\text{event}_1, \ldots, \text{event}_{|E|}\}$ in the documents, where the $k$-th event $e_k = \{t_k, a_k, y_k\}$ consists of triggers $t_k$, arguments $a_k$ (subject $\text{sub}_k$, object $\text{obj}_k$, time $\text{time}_k$, location $\text{loc}_k$) and sentiment polarities $y_k \in \{P, N, O\}$, which represents positive, negative and neutral. We aim to maximize the data likelihood of the training set as follows.

$$
\prod_{k=1}^{|E|} p(e_k) = \prod_{k=1}^{|E|} p(t_k, a_k, y_k) = \prod_{k=1}^{|E|} p(t_k|d) p(a_k|y_k, t_k) p(y_k|t_k)
$$

### 4.1 Contextualized Word Embedding

In the word embedding module, we map each word $x_i$ in the input sequence $d$ into a continuous vector space. Contextualized embedding produced by pre-trained language models [4] have been proved to be capable of improving the performance of a variety of tasks. Here, we employ the contextualized representations produced by BERT-base to obtain the word embedding. Specifically, we input the document $[\{\text{CLS}, w_1, w_2, \ldots, w_m, \text{SEP}\}]$ into BERT-base. Then we obtain the word embeddings $\{x_1^w, x_2^w, \ldots, x_m^w\}$, where $m$ is the number of the words in $d$, where $\{\text{CLS}\}$ is BERT’s special classification token, $\{\text{SEP}\}$ is the special token to denote separation.

### 4.2 Feature-Enhanced Trigger Extractor

Trigger extractor aims to identify whether words trigger an event. We formulate trigger extraction as a token-level classification task and extract all the triggers simultaneously. We integrate the semantic features (e.g., POS and NER) into text modeling because they are useful for this task. For example, most of the triggers are verbs, and most of the arguments are entities and nouns. Stanza [25] is used to obtain the POS and NER tags of the words. We forward the concatenation of three types of embedding, including word embedding $x^w$, pos embedding $x^{pos}$ and ner embedding $x^{ner}$ to a feed-forward network (FFN), $x^f_i = \text{FFN}(\text{concat}(x^w_i, x^{pos}_i, x^{ner}_i))$.

Inspired by [30, 32], we train start and end classifiers to enforce the model focus on the triggers’ boundaries. The distributions of trigger $\text{start}$ and $\text{end}$ are computed as

$$
\begin{align*}
\hat{p}_{i^s}^f &= \text{Sigmoid}(W^f x^f_i + b^f); \\
\hat{p}_{i^e}^f &= \text{Sigmoid}(W^f x^f_i + b^f)
\end{align*}
$$

where $s$ and $e$ denote the start and end indices, $W^f$, $b^f$ and $b^e$ are the learnable weights. As general, we adopt cross entropy (CE) between the predicted probabilities and the ground truth labels as the loss function for fine-tuning.

$$
L_i = \frac{1}{m} \sum_{i=1}^m \text{CE}(\hat{y}_i^f, p_{i^s}^f) + \text{CE}(\hat{y}_i^f, p_{i^e}^f)
$$

where $\hat{y}_i^f / y_i$ is 1 if $i$-th word is the trigger’s start/end.

### 4.3 Trigger-Enhanced Argument Extractor

Argument extractor aims to identify the related arguments concerning the given trigger. To better capture the trigger information, we design a trigger-enhanced argument extractor to integrate the trigger’s representation and position information into word representation.

For the trigger representation, we use its head and tail word representations. Also, we define the word position index according to the relative distance with the trigger. The word position
Table 3: The results of event-level SA with extracted arguments. The best scores are marked with bold.

| Trigger | P  | R  | F1 | Sub | P  | R  | F1 | Obj | P  | R  | F1 | Arguments | P  | R  | F1 | Time | P  | R  | F1 | Loc | P  | R  | F1 | Sentiment | P  | R  | F1 |
|---------|----|----|----|-----|----|----|----|-----|----|----|----|----------|----|----|----|------|----|----|----|-----|----|----|----|-----------|----|----|----|
| ECFEE-O | 41.09 | 21.55 | 32.21 | | 61.40 | 14.73 | 21.99 | 60.71 | 19.30 | 27.59 | 71.90 | 48.46 | 57.89 | 67.78 | 16.12 | 26.05 | 14.60 | 19.24 | 16.60 | 15.93 | 5.83 | 8.54 | 41.69 | 27.59 | 33.21 |
| DCFEE-M | 38.87 | 38.52 | 35.52 | | 34.66 | 19.00 | 24.55 | 40.81 | 25.37 | 31.14 | 38.62 | 59.92 | 59.26 | 67.78 | 16.12 | 26.05 | 14.60 | 19.24 | 16.60 | 15.93 | 5.83 | 8.54 | 41.69 | 27.59 | 33.21 |
| GreedyDec | 67.23 | 36.04 | 50.40 | | 67.78 | 16.12 | 26.05 | 63.74 | 16.62 | 26.26 | 79.08 | 53.30 | 63.68 | 71.90 | 48.46 | 57.89 | 54.79 | 79.08 | 53.30 | 63.68 | 71.90 | 48.46 | 57.89 | 54.79 |
| Doc2EDAG | 38.94 | 21.37 | 31.80 | | 62.11 | 14.03 | 22.89 | 58.75 | 14.03 | 22.65 | 56.25 | 11.89 | 19.64 | 56.25 | 11.89 | 19.64 | 56.25 | 11.89 | 19.64 | 56.25 | 11.89 | 19.64 | 56.25 |
| BERT-QA | 51.40 | 60.85 | 55.73 | | 69.16 | 55.22 | 61.80 | 49.96 | 52.84 | 59.20 | 75.58 | 57.27 | 65.16 | 58.62 | 59.91 | 59.26 | 58.62 | 59.91 | 59.26 | 58.62 | 59.91 | 59.26 | 58.62 |
| ECFEE (Ours) | 54.79 | 79.08 | 53.30 | | 79.08 | 53.30 | 63.68 | 34.66 | 19.00 | 24.55 | 62.11 | 14.03 | 22.89 | 62.11 | 14.03 | 22.89 | 62.11 | 14.03 | 22.89 | 62.11 | 14.03 | 22.89 | 62.11 |

Figure 2: Our E-SA framework.

embedding $x^r_{i+1}$ specific to a trigger $t_k$ can be looked up by
a position embedding matrix, which is randomly initialized and
updated during the training process. Then, we concatenate the trigger’s representation and position embedding with feature-enhanced word representation $x^e_i$, and feed them into FFN module, $x^e_k = \text{FFN}(\text{concat}(x^r_i, x^r_{i+1}, x^f_i, x^f_{i+1}, x^p_{i+1})))$, where $x^r_i$ and $x^r_{i+1}$ is the head and tail representation of the trigger $t_k$ obtained from $x^f_i$.

Similarly, a word $x_i$ is predicted as the start and end of an argument that plays role $r$ w.r.t. $t_k$, with the probability,

$$p_i^r = \text{Sigmoid}(W^e x^e_i + b^e); p_i^{r+1} = \text{Sigmoid}(W^e x^e_{i+1} + b^e)$$

The loss function for argument extraction is,

$$L_a = \frac{1}{M} \sum_{i=1}^{M} \sum_{r=1}^{R} \sum_{c=1}^{C} \text{CE}(y_i^c, p_i^r) + \text{CE}(y_{i+1}^c, p_i^{r+1})$$

where $R$ is the set of roles, including subject, object, time and location.

4.4 Event-Level Sentence Classifier

Besides the trigger, the arguments information can also help to
model the event and its sentiment. Thus, we model the argument role into an embedding $x^r_k$, which tells not only the position but also the type information of the arguments. We integrate it with trigger-enhanced word representation $x^f_k$. Then we adopt a max-pooling layer (MaxPooling) to obtain the event representation, $e^{\text{event}k} = \text{MaxPooling}(\text{concat}(x^r_k, x^f_k))$. We input the event representation $e^{\text{event}k}$ into a softmax layer for sentiment classification, $p_k = \text{Softmax}(W^e e^{\text{event}k} + b^e)$, where $W^e$ and $b^e$ are the learnable parameters. Then, we calculate the loss of SA, $L = \frac{1}{M} \sum_{k=1}^{K} \text{CE}(y_k, p_k)$.

Finally, to learn the relationships among the multiple subtasks, we learn them jointly by adding the losses together, $L = L_t + L_a + L_e$.

5 EXPERIMENTS

5.1 Experimental Setup

Evaluation. As for evaluation, we adopt three metrics: precision (P), recall (R), and F1 scores (F1), the same as [12, 35]. Additionally, we evaluate the performance of sentiment classification with gold arguments in terms of P, R, F1 and accuracy.

Baselines. To verify the effectiveness of our model, we conduct the experiments from two perspectives, end-to-end event extraction and aspect-based SA methods. First, we select four widely-used end-to-end event extraction baselines to verify their performance on structured event-level SA, including DCFEE [31] (consists of two versions: DCFEE-O and DCFEE-M), GreedyDec [36], Doc2EDAG [36], BERT-QA [5]. For a fair comparison, we replace word embeddings with BERT embeddings for DCFEE. Second, we compare our model with four Non-BERT-based and three BERT-based typical ABSA models by inputting the documents and gold events for sentiment prediction, including MemNet [27], ATAE_LSTM [29], MGAN [7], TNet [15], BERT-SPC [4], AEN_BERT [26], LCF-BERT [34]. For the of the limit space, please see the details of the baselines in the related studies.

Implementation Details. BERT [4] is utilized as the word embedding. We use Adam optimizer with the learning rates of 1e-5. The dimensions of position, pos, and ner embedding are 128. The max sequence length is 512. The dropout is 0.1. The reported test results are based on the parameters that obtain the best performance on the development set with five random seeds.

5.2 Main Results

Event-level SA with Extracted Arguments. We apply the typical end-to-end event extraction baselines to structured event-level SA task and report the results of these models and E-SA (Table 3). From this table, we obtain the following findings. First, our model outperforms the strong baselines in most cases. In particular, E-SA obtains better performance than all the baselines significantly in terms of F1 for all the subtasks. Second, E-SA captures the sentiment information of the events effectively. These baselines focus on predicting the event type while the argument information and the relationships among the events are ignored by these models. Third, most of the models can not extract the location of the event since...
We observe that the effectiveness of the components contained in the model will reduce the performance of sentiment classification. To further prove the effectiveness of the components contained in our model, we remove both the trigger and argument information to capture the sentiment information of the events more effectively.

NER can improve the performance effectively because the argument information can also help the model learn the event representations, position embedding) will reduce the performance of each subtask by modeling the relationships among the multi-subtasks and multiple events. Moreover, we integrate the trigger and argument information into sentiment classification to capture the sentiment information towards the given events effectively.

5.3 Ablation Studies
To further prove the effectiveness of the components contained in our model, we do ablation studies (Table 5). First, comparing with the state-of-the-art baselines. Additionally, we label a real-world corpus for this task for lack of the off-the-shelf datasets. It would be interesting to investigate how to integrate users’ reviews to better capture the sentiment information of the events.

**Event-level SA with Gold Arguments.** To further verify the effectiveness of E3SA on inferring the events’ sentiment polarities, we adopt the existing strong baselines of ABSA and perform sentiment classification over structured event-level SA (Table 4). We observe that E3SA outperforms all the baselines in terms of F1 and accuracy. All the baselines focus on the interaction between the event and the text to capture the event-specific sentiments. E3SA not only considers the relationships among multiple subtasks but also the relationships among multiple events. Moreover, we integrate the trigger and argument information into sentiment classification to capture the sentiment information towards the given events effectively.

6 CONCLUSIONS AND FUTURE WORK
In this paper, we propose an effective E3SA approach for structured event-level sentiment analysis. This joint approach models the relationships among the multi-subtasks and multi-events with structured arguments. We conduct extensive experiments to evaluate our model on both event extraction and sentiment classification. The results demonstrate the great advantages of our model by comparing it with the state-of-the-art baselines. Additionally, we label a real-world corpus for this task for lack of the off-the-shelf datasets. It would be interesting to investigate how to integrate users’ reviews to better capture the sentiment information of the events.

**ACKNOWLEDGMENTS**
The authors wish to thank the reviewers for their helpful comments and suggestions. This research is funded by the Science and Technology Commission of Shanghai Municipality (No. 19511120200&21511100100 and 21511100402) and by Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University, No. 2020KEY001 and the Fundamental Research Funds for the Central Universities. This research is also funded by the National Key Research and Development Program of China (No. 2021ZD0114002), the National Nature Science Foundation of China (No. 61906045), and Shanghai Science and Technology Innovation Action Plan International Cooperation project “Research on international multilanguage online learning platform and key technologies (No.20510780100)”. The computation is performed in ECNU Multi-functional Platform for Innovation (001).

**REFERENCES**

[1] Rushlene Kaur Bakshi, Navneet Kaur, Ravneet Kaur, and Gurpreet Kaur. 2016. Opinion mining and sentiment analysis. In Proceedings of INDIACom. 452–455.
[2] Yubo Chen, Lihegui Xu, Kang Liu, Diaoqian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In Proceedings of ACL ’16–176.
[3] Lingxia Deng and Yanjie Wiebe. 2015. Joint prediction for entity/event-level sentiment analysis using probabilistic soft logic models. In Proceedings of EMNLP. 179–189.
[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of NAACL-HLT.
[5] Xi Xiong and Claire Cardie. 2020. Event Extraction by Answering (Almost) Natural Questions. In Proceedings of EMNLP. 671–683.
[6] Monireh Ebrahimi, Amir Hossein Yazdavar, and Amin Sheth. 2017. Challenges of sentiment analysis for dynamic events. IEEE Intelligent Systems. 32, 5 (2017), 70–75.
[7] Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In Proceedings of EMNLP. 3433–3442.
[8] Fuli Feng, Moxin Li, Cheng Luo, Ritchie Ng, and Tat-Seng Chua. 2021. Hybrid Learning to Rank for Financial Event Ranking. In Proceedings of SIGIR. 233–243.
[9] Tomohiro Fujikura, Hiroshi Nakagawa, and Toyooki Nishida. 2007. Understanding of sentiment of People from News Articles: Temporal Sentiment Analysis of Social Events. In ICWSM.
[10] Rajkumar S Jagdale, Vishal S Shirasat, and Sachin N Deshmukh. 2016. Sentiment analysis of events from Twitter using open source tool. International Journal of Computer Science and Mobile Computing. 5, 4 (2016), 475–485.
[11] Klaus Krippendorff. 2011. Computing Krippendorff’s alpha-reliability. (2011).
[12] Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In Proceedings of ACL. 73–82.
[13] Quanzhi Li and Qiong Zhang. 2020. A Unified Model for Financial Event Classification, Detection and Summarization. In Proceedings of IJCAI. Christian Bessiere (Ed.). 4668–4674.
[14] Sha Li, Heng Ji, and Jiawei Han. 2021. Document-Level Event Argument Extraction by Conditional Generation. In Proceedings of NAACL. 894–908.
[15] Xin Li, Lidong Bing, Wai Lam, and Bei Shui. 2018. Transformation Networks for Target-Oriented Sentiment Classification. In Proceedings of ACL 946–956.

[16] Junsui Liao, Xiang Zhao, Xinyi Li, Lingling Zhang, and Jueyou Tang. 2021. Learning Discriminative Neural Representations for Event Detection. In Proceedings of SIGIR 644–653.

[17] Bing Liu. 2012. Sentiment analysis and opinion mining. Synthesis lectures on human language technologies 5, 1 (2012), 1–167.

[18] Xiao Liu, Zhuochen Luo, and He-Yan Huang. 2018. Jointly Multiple Events Extraction via Attention-based Graph Information Aggregation. In Proceedings of EMNLP 1247–1256.

[19] Masoud Makrehchi, Sameena Shah, and Wenhui Liao. 2013. Stock prediction using event-based sentiment analysis. In Proceedings of WI-IAT, Vol. 1. 337–342.

[20] Thien Huu Nguyen, Kyoung Joong Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In Proceedings of NAACL. 300–309.

[21] Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In Proceedings of ACL. 365–371.

[22] Mamta Patil and HK Chavan. 2018. Event based sentiment analysis of Twitter data. In Proceedings of ICCCIC. 1050–1054.

[23] Alexandru Petrescu, Ciprian-Octavian Trucă, and Elena-Simona Apostol. 2019. Sentiment Analysis of Events in Social Media. In 2019 IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP). 143–149.

[24] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In Proceedings of SemEval 2014. 27–35.

[25] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. 2020. Stanzia: A Python Natural Language Processing Toolkit for Many Human Languages. In Proceedings of ACL: Demo. 101–108.

[26] Youwei Song, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Ex- ploiting Pre-trained Language Models for Event Extraction and Generation. In Proceedings of ACL. 5284–5294.

[27] Yiying Yang, Zhongyu Wei, Qin Chen, and Libo Wu. 2019. Using External Knowledge for Financial Event Prediction Based on Graph Neural Networks. In Proceedings of CIKM. 2161–2164.

[28] Tongtao Zhang, Heng Ji, and Avirup Sil. 2019. Joint entity and event extraction via attention-based graph information aggregation. In Proceedings of ACL. 365–371.

[29] Yiying Yang, Zhongyu Wei, Qin Chen, and Libo Wu. 2019. Doc2EDAG: An end-to-end document-level framework for Chinese financial event extraction. arXiv:1904.07535 (2019).

[30] Jie Zhou, Jimmy Xiangji Huang, Qin Chen, Qimin Vivian Hu, Tingting Wang, and Liang He. 2019. Deep learning for aspect-level sentiment classification: survey, vision, and challenges. IEEE access 7 (2019), 78454–78483.

[31] Jie Zhou, Jimmy Xiangji Huang, Qimin Vivian Hu, and Liang He. 2020. Modeling multi-aspect relationship with joint learning for aspect-level sentiment classification. In International Conference on Database Systems for Advanced Applications. Springer, 786–802.

[32] Jie Zhou, Jimmy Xiangji Huang, Qimin Vivian Hu, and Liang He. 2020. SK-GCN: modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification. Knowledge-Based Systems 205 (2020), 106292.

[33] Jie Zhou, Junfeng Tian, Rui Wang, Yuanbin Wu, Wenming Xiao, and Liang He. 2020. Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis. In Proceedings of the 28th International Conference on Computational Linguistics. 568–579.

[34] Yiying Yang, Zhongyu Wei, Qin Chen, and Libo Wu. 2019. Using External Knowledge for Financial Event Prediction Based on Graph Neural Networks. In Proceedings of CIKM. 557–562.

[35] Xujian Zhou, Xiaohui Tao, Jianming Yong, and Zhenyu Yang. 2013. Sentiment analysis on tweets for social events. In Proceedings of CSCWD. 557–562.