Joint Extraction of Financial Events Using Multi-layer Convolutional Neural Networks: Postprint

Authors: Li Xuhui, Cheng Wei, Tang Xiaoya, Yu Tao, Chen Zhuang, Qian Tieyun

Date: 2023-04-01T00:00:00+00:00

Abstract

[Purpose/Significance] To further enhance the performance of event extraction in the financial domain and strengthen the interconnection between its two constituent subtasks. [Method/Process] This study investigates event extraction on Chinese financial texts and proposes a joint financial event extraction framework that integrates a pre-trained model with a multi-layer convolutional neural network. The pre-trained BERT model is first employed to capture comprehensive semantic representations of sentence sequences, which are subsequently processed by the MultiCNN architecture designed in this work to hierarchically extract features from local windows and high-dimensional semantic spaces, thereby simultaneously accomplishing event identification and argument extraction. Furthermore, the introduction of contrastive loss further reinforces the association between these two tasks. [Results/Conclusion] The proposed method achieves an F1 score of 82.20% on a Chinese financial event dataset, demonstrating consistent improvements over various baseline extraction models.

Full Text

A Joint Extraction Method for Financial Events Based on Multi-Layer Convolutional Neural Networks

Li Xuhui[1,2], Cheng Wei[1], Tang Xiaoya[3], Yu Tao[1], Chen Zhuang[3], Qian Tieyun[3]

[1]School of Information Management, Wuhan University, Wuhan 430072
[2]Institute of Big Data, Wuhan University, Wuhan 430072
[3]School of Computer Science, Wuhan University, Wuhan 430072

Abstract:
[**Purpose/Significance**] To further improve the effectiveness of event extrac-

tion in the financial domain and enhance the correlation between its two subtasks. [**Method/Process**] This paper conducts research on event extraction from Chinese financial texts and proposes a joint financial event extraction method that integrates a pre-trained model with a multi-layer convolutional neural network. First, the pre-trained BERT model captures comprehensive semantic information from sentence sequences, which is then fed into the MultiCNN architecture designed in this paper to hierarchically extract local window and high-dimensional spatial semantic information, simultaneously accomplishing both event recognition and element extraction. Contrastive loss is further introduced to strengthen the association between the two tasks. [**Result/Conclusion**] The F1 score reaches 82.20% on a Chinese financial event dataset, representing a certain improvement over various benchmark extraction models.

Keywords: Chinese event extraction; convolutional neural network; pretraining model; joint learning

1 Introduction

Events represent a semantic unit through which people express information, opinions, or facts, generally referring to objective occurrences where specific individuals or objects interact at particular times and locations. They constitute a fundamental way of understanding the world [1]. The financial industry generates massive domain-specific textual data, including news, announcements, and social media posts, which contain high-timeliness, high-value information represented by financial events such as investment/financing activities and mergers & acquisitions. These events can assist enterprises in information-driven decision-making and provide foundational technical support for constructing financial event systems and knowledge graphs, holding significant application value. How to effectively obtain such event information has long been a shared concern in both industry and academia, with event extraction from domain texts being the primary approach.

Event extraction involves identifying specific event types from natural language text while extracting associated temporal, locational, and participant elements, ultimately presenting them in a structured format [2]. Typically, event extraction comprises two subtasks: (1) event detection and classification, and (2) event element identification and classification. As a hot research topic in natural language processing, event extraction has attracted widespread academic attention in recent years and plays an important role in information extraction, public opinion analysis, and knowledge base construction. Major evaluation conferences such as ACE (Automatic Content Extraction) have dedicated tracks for event extraction. ACE defines 33 event types spanning diverse domains including births/deaths, company establishment, traffic accidents, and imprisonment/release, providing parallel annotated corpora in multiple languages for evaluation [3]. Most current research optimizes and experiments on the ACE 2005 corpus.

Financial event extraction aims to rapidly and accurately extract key event information from financial texts according to business requirements. Financial event extraction exhibits distinct domain-specific characteristics, primarily reflected in relatively specialized event types and high accuracy requirements for element identification. Directly transplanting traditional event extraction methods faces several challenges: (1) Sparse distribution of financial event categories—the ACE 2005 Chinese corpus contains only 633 documents covering 33 event types, with insufficient training data and many categories irrelevant to finance (e.g., death, attack, demonstration), while excessive categories cause severe overfitting in extraction models; (2) Insufficient accuracy in event sentence classification and element extraction, requiring further model optimization; and (3) Independent modeling of the two subtasks, leading to error propagation between them. Therefore, financial event extraction demands more targeted corpus accumulation and extraction methods.

For these reasons, this paper focuses on the practical needs of financial event extraction, explores key technical issues, and proposes a novel financial event extraction method. The main contributions are threefold: (1) Establishing a financial domain event classification system, collecting real news corpora from financial information websites, and performing character-level BIO annotation according to event extraction requirements; (2) Proposing BERT-MultiCNN, a joint event extraction method integrating the pre-trained language model BERT [4] with multi-layer convolutional neural networks [5] to hierarchically capture event semantic information in sentences and further improve extraction effectiveness; and (3) Introducing a two-stage joint learning mechanism to address multi-task collaboration and resolve error propagation in pipeline models, establishing reasonable associations between tasks through contrastive loss to enable error correction in the first-stage event classification using information from the second-stage element extraction. Experimental results demonstrate that our model achieves an F1 score of 82.20% on a Chinese financial event dataset, further improving the effectiveness of both financial event extraction subtasks and demonstrating high application value.

2 Related Research

Event extraction is a critical task in natural language processing with broad applications in public opinion analysis, semantic understanding, and text summarization. Extensive research has been conducted by scholars both domestically and internationally. Current event extraction research can be broadly categorized into three approaches: pattern matching methods, feature-engineered machine learning methods, and deep learning methods.

Pattern matching methods typically involve designing standardized extraction rules or templates with the assistance of domain experts, with pattern construction being the core component. Examples include E. Riloff' s AutoSlog [6] and its improved version AutoSlog-TS [7], which utilize pattern dictionaries to build templates for automatic extraction and can learn new patterns from existing ones, significantly reducing the cost of manual rule construction to a certain extent. In the financial domain, R. Feldman et al. [8] constructed a financial sentiment dictionary to achieve phrase-level sentiment polarity matching and formulated event extraction rules; Luo Ming et al. [9] defined a financial event representation model, used word embedding tools to automatically generate synonyms, and implemented multi-type financial event extraction based on rule patterns. In summary, pattern matching methods require extensive standardized extraction rules, entail high construction costs, and lack generality, thus having significant limitations in application.

With the widespread success of machine learning in speech and translation, researchers gradually turned to machine learning-based extraction. This approach constructs multiple features including lexical, syntactic, and positional information as input to machine learning models for automated event extraction, with common models including Support Vector Machines (SVM), Conditional Random Fields (CRF), and Hidden Markov Models (HMM). Li Xiang and Yang Xiaolin et al. [10] designed three types of features—lexical, syntactic, and semantic—and utilized SVM for news event classification. L. Hou et al. [11] employed CRF for machine learning model construction, integrating five feature types (lexical, semantic, dependency relations, syntactic relations, and relative position) with semantic role features to achieve event extraction. The challenge of machine learning methods lies in feature selection and combination, with research differences primarily reflected in these choices, inevitably introducing subjective bias.

Unlike traditional machine learning, deep learning approaches represent text sequences as computable multi-dimensional tensors and build end-to-end deep learning models for event trigger and argument classification. Event sentence classification is equivalent to k-class classification (where k equals event sentence categories + 1, with the extra class representing non-event sentences), while event element extraction requires multi-class classification with t categories at each position (where t equals all event element categories + 1, with the extra class representing non-element labels). This approach typically eliminates the need for feature engineering, reducing subjective influence on model input. Commonly used feature extractors include Convolutional Neural Networks (CNN [5,12-15]), Recurrent Neural Networks (RNN [13-17]), Transformer encoders [18-22], and Graph Neural Networks [19,23-24]. For instance, Y. Chen et al. [12] first applied deep neural networks to event extraction, designing a dynamic multi-pooling CNN that integrated word embeddings, position vectors, and entity type vectors to accomplish both event classification and element extraction. T.H. Nguyen et al. [16] utilized Bi-GRU for sentence encoding and designed different memory matrices for joint extraction of event classification and element identification, reducing error propagation. S. Zheng et al. [19] proposed Doc2EDAG, an end-to-end solution for arguments scattered across multiple sentences, generating entity-based directed acyclic graphs encoded with multiple Transformers. At the graph level, many scholars treat sentence sequences as directed acyclic graphs and employ graph neural networks for event classification and extraction, such as T.H. Nguyen et al. [23] who proposed a graph convolutional network (GCN) based on syntactic dependency trees for event detection, later improved by S. Cui et al. [24] with a relation-aware GCN (RA-GCN) structure that studied the impact of different syntactic relation labels on event extraction.

A comparative analysis of these deep learning methods for event extraction is presented in Table 1 .

Applying machine and deep learning to financial event extraction has been a key focus in both academia and industry. Beyond model exploration, financial event extraction involves two application aspects: (1) Corpus acquisition—supervised methods require mature annotated corpora. H. Yang et al. [25] obtained more financial domain annotated data through distant supervision and utilized BiLSTM-CRF for sentence-level and document-level event extraction. L. Ein-Dor et al. [26] extracted weak labels from Wikipedia data for financial events and demonstrated effectiveness through experiments. Z. Zhou et al. [27] proposed a two-tier event detection model for document-level corporate financial events and developed a new dataset EDT based on news corpora for corporate event identification and stock price prediction; (2) Output results—financial event extraction requires clear application objectives. S. Rönqvist et al. [28] used deep learning to extract financial risk events from news and generated analysis reports on banking crisis indices and government interventions. S. Carta et al. [29] employed hierarchical clustering to classify financial events, providing users with structured daily news summaries while alerting them to potentially important events. L.D. Corro et al. [30] designed a unique attention network to process large volumes of news headlines for predicting stock price movements (up/down/stable), achieving good prediction results without labeled data.

Current research on pre-trained model-based event extraction largely delegates semantic capture to general-purpose pre-trained models, lacking specificity for capturing local information in individual sentences. Convolutional Neural Networks (CNN) utilize kernel sharing mechanisms to fully leverage contextual information within local windows for feature extraction, achieving excellent results in text classification [5,31], sequence labeling [32-34], and event extraction [12-14,35]. However, applying these studies to event extraction presents several issues: (1) Feature extraction—most previous research stacks word embeddings with lexical and syntactic features for CNN-based extraction and integration, but CNN’ s shallow network structure limits semantic encoding capacity; (2) CNN structure—the convolutional architecture is overly simplistic, with optimization limited to adjusting layer numbers or improving pooling methods; (3) Using CNNs independently for event classification and element extraction causes error propagation between the two tasks.

Therefore, this paper combines the advantages of pre-trained models and CNNs, first using the BERT pre-trained language model for general semantic extraction, then designing a unique multi-layer CNN to hierarchically capture local information for joint end-to-end training of both tasks, ultimately achieving event sentence classification and element extraction.

3 Financial Event Joint Extraction Method

This paper proposes a financial event joint extraction method that effectively addresses error propagation in event extraction. To obtain typical financial events, we first constructed a financial event classification system. For model optimization and effectiveness validation, we selected three key event types—mergers & acquisitions, strategic cooperation, and investment/financing—for corpus annotation and conducted related optimization and exploratory experiments.

3.1 Financial Event Classification System

Based on background research in the financial domain, this paper initially established a financial event classification system from a corporate finance perspective. The complete event categories and subcategory names are shown in Table 2 . To obtain extensive extraction corpora for specific event categories, we used keyword extraction technology to extract event feature words and employed Word2Vec [36] for feature word expansion, collecting large volumes of Chinese event extraction corpora based on the expanded feature word sets.

3.2 Problem Definition

Financial event extraction is generally defined as: automatically extracting key events and their associated elements (e.g., time, location, event subject) from unstructured text to obtain structured event representations. As shown in the example below, the model must classify the sentence as a strategic cooperation event and extract the event trigger ("strategic cooperation"), cooperating companies ("Haitian Water Affairs Group", "CITIC Bank Chengdu Branch"), event time ("April 6"), and event location ("Chengdu"). The character-level label mapping is detailed in Table 3 .

Example: "On April 6, Haitian Water Affairs Group and CITIC Bank Chengdu Branch signed a strategic cooperation agreement in Chengdu."

For the two subtasks of event sentence classification and event element identification, there are two primary processing approaches: pipeline and joint extraction. Pipeline methods first detect event types then extract corresponding elements from event sentences, while joint extraction accomplishes both tasks simultaneously. Due to error propagation in pipeline architectures—where errors in the first-stage event sentence classification affect subsequent element identification—this paper adopts a joint extraction approach. Specifically, we treat event sentence identification as a sentence classification task and event trigger/element identification as a character-level sequence labeling task, building an end-to-end deep learning model for event extraction.

3.3 Method Overview

To accomplish the two subtasks of financial event extraction—event classification and element extraction—this paper proposes BERT-MultiCNN, a joint financial event extraction method based on multi-layer convolutional neural networks. First, BERT' s powerful semantic encoding capability vectorizes the sentence sequence, obtaining hidden vector representations H_1 for each token position and sentence vector representation C_1. H_1 is then processed through a multi-layer CNN architecture to further extract local semantic information, yielding vector representation H_2. H_2 is fed into a fully connected classification model to obtain label distributions for each position, followed by a CRF layer to control label output and produce final trigger and element extraction results. For joint extraction, we design a unified end-to-end architecture to align optimization directions for both tasks. Additionally, while performing sequence labeling for the second task, we apply pooling to H_2 to obtain sentence vector C_2, which is fused with C_1 for event classification results and used to compute contrastive loss, forming a joint optimization objective. The overall architecture is shown in Figure 1 [Figure 1: see original paper], consisting of the input layer, BERT encoding layer, MultiCNN layer, fully connected layer, CRF layer, and contrastive loss between pre- and post-convolution sentence representations.

Our model' s innovations include: (1) Combining the advantages of pre-trained models and traditional CNN architectures for the two extraction tasks at sentence and token levels; (2) Introducing a joint extraction paradigm to resolve error propagation; and (3) Incorporating sentence vector contrastive learning to optimize multi-task collaboration.

3.3.1 BERT Encoding Layer This paper employs Google' s pre-trained BERT model [4] as the word vector encoder. BERT' s main building block is the Transformer [18] encoder, which stacks multiple bidirectional Transformer layers to capture bidirectional deep semantic information in sentence sequences. BERT's input vector comprises three components: Token Embedding (word vectors using one-hot encoding or public embeddings like Word2Vec [36] or GloVe [34], typically character-level for Chinese), Position Embedding (relative position encoding for each character), and Segment Embedding (sentence separation vectors). The input vector is the sum of these three components:

$$E_{input} = E_{token} + E_{position} + E_{segment} \tag{1}$$

After obtaining BERT input, it enters the Transformer encoder, which consists primarily of multi-head self-attention and feed-forward neural networks. Self-attention learns relationships between all position pairs, with "multi-head" enabling multiple perspectives. Details are shown in equations (2)-(4):

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{3}$$

$$MultiHead(Q,K,V) = Concat(head_1, head_2, ..., head_h)W^O \tag{4}$$

The feed-forward network comprises two linear transformations with ReLU activation:

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \tag{5}$$

Residual connections and layer normalization (Add & Norm) after self-attention and feed-forward networks facilitate training and prevent gradient vanishing. After BERT encoding, we take the last layer' s output as the final representation, where each position corresponds to a 768-dimensional hidden vector. The initial [CLS] token represents the sentence classification vector (C_1) after a fully connected layer:

$$H_1 = BERT(E_{input}) \tag{6}$$

$$C_1 = FC_1(H_1[0,:,:]) \tag{7}$$

3.3.2 MultiCNN Layer Convolutional Neural Networks (CNN) are specialized deep feedforward networks originally for computer vision, inspired by human visual systems' receptive fields. They have been successfully adapted for text processing tasks like classification and sequence labeling [5,12-13,32]. Typically, CNNs combine convolutional, pooling, and activation layers.

For trigger and element identification, we treat the task as character-level sequence labeling using B, I, O tags to mark entity beginnings, interiors, and non-entities, enabling simultaneous classification of all positions. We apply 1D convolution kernels sliding along the sentence sequence to capture semantic and contextual window features for each character. To maintain sequence length, we design padding operations for different kernel sizes. The feature extraction process for a kernel size of 3 is shown in Figure 2 [Figure 2: see original paper].

Specifically, our approach involves two stages: (1) Local window convolution using parallel kernels of sizes 3 and 5, with feature map stacking; (2) Semantic enhancement via three additional convolutional layers with kernel size 5 to extract high-level semantic space information. The final convolutional output enters a fully connected layer for dimension transformation to obtain label classification probabilities P_{LABEL}:

$$H_2 = MultiCNN(H_1) \tag{8}$$

$$P_{LABEL} = FC(H_2) \tag{9}$$

3.3.3 CRF Layer Trigger and element identification can be treated as label classification tasks, but traditional softmax classifiers ignore label dependencies, causing label bias issues. Conditional Random Fields (CRF) model dependencies between sequence labels, proving effective for POS tagging, segmentation, and NER [39-40]. CRF jointly models the entire label sequence. For input sequence X = {x_1, x_2, ···, x } and predicted label sequence y = {y_1, y_2, ···, y }, the overall score combines node scores (label probabilities) and path scores (label transition probabilities):

$$S(X, y) = \sum_{i=1}^{m} P_{i,y_i} + \sum_{i=0} T_{y_i,y_{i+1}} \tag{10}$$

The training objective maximizes the correct path score. After global normalization across all paths, we obtain the normalized probability:

$$P(y^*|X) = \frac{e^{S(X,y^*)}}{\sum_{\tilde{y}\in Y_X} e^{S(X,\tilde{y})}} \tag{11}$$

The optimization target is the log-likelihood log P(y*|X). During prediction, Viterbi decoding on label sequence probabilities yields the final labels.

3.3.4 Training Loss For event sentence classification, we use cross-entropy loss to measure the gap between true and predicted distributions. Let Q_1 be the true distribution and P_{sentence} the predicted distribution:

$$P_{sentence} = softmax(C_1) \tag{12}$$

$$Loss_1 = H(P_{sentence}, Q_1) = -\sum Q_1(x) \log_2 P_{sentence}(x) \tag{13}$$

For element identification, we define $Loss_2$ as the CRF layer' s negative log-likelihood:

$$Loss_2 = -\log P(y^*|X) \tag{14}$$

To associate the two subtasks during joint learning, we combine their losses for information sharing and further fuse semantic information through contrastive loss Loss_3 between pre- and post-convolution sentence vectors. Specifically, we apply mean pooling to H_2 to obtain C_2 and define contrastive loss as the cross-entropy between C_1 and C_2. Intuitively, minimizing the distance between convolutional representations aligns both tasks' learning directions:

$$Loss_3 = -\sum C_2(x) \log_2 C_1(x) \tag{15}$$

The joint training loss is a weighted sum of the three components, with weights tuned experimentally:

$$Loss = Loss_1 + Loss_2 + 0.5 \times Loss_3 \tag{16}$$

4 Experiments and Analysis

4.1 Data Source

Our financial event extraction corpus is self-constructed. We crawled original document-level data from financial news websites Caixin (https://www.caixin.com/) and NBD (http://www.nbd.com.cn/) using Python. After sentence segmentation, deduplication, and stopword removal, we annotated event sentences, triggers, and elements using the open-source brat system (http://brat.nlplab.org/). We adopted BIO character-level annotation, where B marks entity beginnings, I marks entity interiors, and O marks non-entities. For example, the sentence "On April 6, Haitian Water Affairs Group and CITIC Bank Chengdu Branch signed a strategic cooperation agreement in Chengdu" receives labels "1 B-time I-time I-time I-time B-cpn I-cpn I-cpn I-cpn I-cpn I-cpn O B-cpn I-cpn I-cpn I-cpn I-cpn I-cpn I-cpn I-cpn I-cpn O B-loc I-loc O O B-trig I-trig I-trig I-trig O O", where the initial "1" indicates event type (0 for non-event). To minimize annotation errors, we conducted unified training and implemented a two-person parallel annotation plus one-person review process. The final annotated corpus statistics are shown in Table 4 , split 8:1:1 for training, validation, and test sets.

4.2 Evaluation Metrics and Environment

We evaluate models using Precision (P), Recall (R), and their harmonic mean F1. An extraction is considered correct only when both event sentence classification and corresponding element extraction are accurate. Experimental environment configurations are listed in Table 5 , and deep learning model parameters in Table 6 .

4.3 Baseline Model Comparison

To validate our model' s effectiveness, we compare against: (1) BiLSTM-CRF [40], a classic sequence labeling model; (2) BERT-BiLSTM-CRF, using BERT for semantic extraction followed by BiLSTM; (3) IDCNN-CRF [41-42], employing dilated convolutions to expand receptive fields; and (4) BERT-IDCNN-CRF [43], combining BERT with dilated convolutions. Comparison results are shown in Table 7 .

Results demonstrate: (1) BERT significantly improves both RNN and CNN architectures by effectively modeling contextual information through large-scale pre-training, with improvements of +18.97% for BiLSTM and +15.40% for IDCNN, indicating BiLSTM' s greater potential with pre-training; (2) BiLSTM maintains natural advantages for sentence sequence modeling over IDCNN, as dilated convolutions' fixed-width perception cannot properly capture specific relationships between event elements; (3) Our BERT-MultiCNN achieves the best performance, improving F1 by 2.59% over the previous best BERT-BiLSTM-CRF, demonstrating its effectiveness for the joint extraction task.

4.4 Ablation Study

To analyze each module' s contribution, we conduct ablation experiments shown in Table 8 . Results reveal: (1) Adding the multi-layer CNN structure improves F1 by 1.88%, confirming its enhancement of semantic capture from the encoding layer; (2) Incorporating CRF improves precision (+1.57%) and recall (+1.87%), validating its effectiveness for modeling element label dependencies; (3) Removing contrastive loss causes recall to drop sharply by 4.15% while precision remains similar, proving that contrastive loss leverages first-stage classification information to identify more event entities, beneficial for high-recall scenarios.

4.5 Results Discussion

Experiments confirm that our multi-layer CNN-based joint extraction method can rapidly and accurately extract structured event information from raw news text without auxiliary information. Extraction examples are shown in Table 9 .

To further demonstrate our model' s improvement on two-stage error propagation, we compare subtask performance across models in Table 10 . Traditional models' element extraction results heavily depend on first-stage classification accuracy—when event classification is wrong (e.g., BiLSTM-CRF, IDCNN-CRF), subsequent extraction becomes meaningless. Our BERT-MultiCNN joint method maintains extraction accuracy while using second-stage element information to correct classification errors, achieving overall optimal performance through joint learning.

5 Conclusion

This paper focuses on financial event extraction' s two core subtasks—event classification and element extraction—addressing insufficient CNN utilization, limited pre-trained semantic extraction, and inter-subtask error propagation. We propose BERT-MultiCNN, a joint extraction framework that uses BERT for primary semantic feature extraction applied to event classification, processes character-level encodings through multi-layer CNNs for hidden representations used in element extraction, and strengthens subtask associations through sentence vector contrastive learning. Experiments demonstrate state-of-the-art performance on financial event extraction, applicable to both financial and other domains.

Our current joint extraction model primarily handles single-event extraction. Future work will explore incorporating document-level information to meet the requirements of complex multi-event extraction scenarios.

References

[1] AHN D. The stages of event extraction[C]//Proceedings of the workshop on annotating and reasoning about time and events. USA: Association for Computational Linguistics, 2006: 1-8.

[2] CHEN C, NG V. Joint modeling for chinese event extraction with rich linguistic features[C]//Proceedings of COLING 2012. Mumbai: The COLING 2012 Organizing Committee, 2012: 529-544.

[3] DODDINGTON G, MITCHELL A, PRZYBOCKI M, et al. The automatic content extraction (ace) program-tasks, data, and evaluation[J]. Proc Lrec, 2004, 2(1): 837-840.

[4] DEVLIN J, CHANG M, LEE K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[C]//Proceedings of the North American Chapter of the Association for Computational Linguistics: human language technologies. Minneapolis: The NAACL-HLT Press, 2019: 4171-4186.

[5] KIM Y. Convolutional neural networks for sentence classification[C]//Proceedings of the 2014 conference on empirical methods in natural language processing. Doha: Association for Computational Linguistics, 2014: 1746-1751.

[6] RILOFF E. An empirical study of automated dictionary construction for information extraction in three domains[J]. Artificial intelligence, 1996, 85(1/2): 101-134.

[7] RILOFF E. Automatically generating extraction patterns from untagged text[C]//Proceedings of the national conference on artificial intelligence. Oregon: Association for the Advancement of Artificial Intelligence, 1996: 1044-1049.

[8] FELDMAN R, ROSENFELD B, BAR-HAIM R, et al. The stock sonar-sentiment analysis of stocks based on a hybrid approach[EB/OL]. [2021-11-10].

https://www.researchgate.net/publication/221016483_{{{The}}{{Stock}}}{Sonar}-{{{Sentiment}}{{Analysis}}}{{{of}}{{Stocks}}}{{{Based}}{{on}}}a_{{{Hybrid}}_{{Approach}}}.

[9] 罗明, 黄海量. 基于词汇-语义模式的金融事件信息抽取方法 [J]. 计算机应用, 2018, 38(01): 84-90.

[10] 李响, 杨小琳, 魏勇, 等. 基于支持向量机的新闻事件类型识别 [J]. 地理信息世界, 2019, 26(02): 73-78.

[11] HOU L, LI P, ZHU Q, et al. Event argument extraction based on CRF[C]//Proceedings of the 13th chinese conference on chinese lexical semantics. Berlin: Springer, 2012: 32-39.

[12] CHEN Y, XU L, LIU K, et al. Event extraction via dynamic multi-pooling convolutional neural networks[C]//Proceedings of the 53rd annual meeting of the Association for Computational Linguistics and the 7th international joint conference on natural language processing (volume 1: long papers). Beijing: Association for Computational Linguistics, 2015: 167-176.

[13] ZENG Y, YANG H, FENG Y, et al. A convolution biLSTM neural network model for chinese event extraction[C]//NLPCC-ICCPOL 2016. Kunming: Springer, 2016: 275-287.

[14] 陈斌, 周勇, 刘兵. 基于卷积双向长短期记忆网络的事件触发词抽取 [J]. 计算机工程, 2019, 45(01): 153-158.

[15] 吴文涛, 李培峰, 朱巧明. 基于混合神经网络的实体和事件联合抽取方法 [J]. 中文信息学报, 2019, 33(08): 77-83.

[16] NGUYEN T H, CHO K, GRISHMAN R. Joint event extraction via recurrent neural networks[C]//Proceedings of the 2016 conference of the North American Chapter of the Association for Computational Linguistics: human language technologies. San Diego: Association for Computational Linguistics, 2016: 300-309.

[17] 陈斌. 基于长短期记忆网络的事件抽取研究与应用 [D]. 徐州: 中国矿业大学, 2019.

[18] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//NIPS' 17. New York: Curran Associates, 2017: 6000-6010.

[19] ZHENG S, CAO W, XU W, et al. Doc2EDAG: An end-to-end document-level framework for chinese financial event extraction[C]//Proceedings of the 57th annual meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019: 337-346.

[20] YANG S, FENG D, QIAO L, et al. Exploring pre-trained language models for event extraction and generation[C]//Proceedings of the 57th annual meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019: 5284-5294.

[21] ZHAO L, LI L, ZHENG X. A BERT based sentiment analysis and key entity detection approach for online financial texts[C]//Proceedings of the 2021

IEEE 24th international conference on computer supported cooperative work in design. 2021: 1233-1238.

[22] DU X, CARDIE C. Event extraction by answering (almost) natural questions[C]//Proceedings of the 2020 conference on empirical methods in natural language processing. Online: Association for Computational Linguistics, 2020: 671-683.

[23] NGUYEN T H, GRISHMAN R. Graph convolutional networks with argument-aware pooling for event detection[C]//AAAI. Louisiana: Association for the Advancement of Artificial Intelligence, 2018: 5900-5907.

[24] CUI S, YU B, LIU T, et al. Event detection with relation-aware graph convolutional networks[J]. CoRR, 2020, abs/2002.10757.

[25] YANG H, CHEN Y, LIU K, et al. DCFee: A document-level chinese financial event extraction system based on automatically labeled training data[C]//Proceedings of ACL 2018, System Demonstrations. Melbourne: Association for Computational Linguistics, 2018: 50-55.

[26] EIN-DOR L, GERA A, TOLEDO-RONEN O, et al. Financial event extraction using wikipedia-based weak supervision[J]. ArXiv, 2019, abs/1911.10783.

[27] ZHOU Z, MA L, LIU H. Trade the event: corporate events detection for news-based event-driven trading[C]//Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Online: Association for Computational Linguistics, 2021: 2114-2124.

[28] RÖNNQVIST S, SARLIN P. Bank distress in the news: describing events through deep learning[J]. Neurocomputing, 2017, 264: 57-70.

[29] CARTA S, CONSOLI S, PIRAS L, et al. Event detection in finance using hierarchical clustering algorithms on news and tweets[J]. PeerJ computer science, 2021, 7: e438.

[30] CORRO L D, HOFFART J. Unsupervised extraction of market moving events using neural attention[J]. ArXiv, 2020, abs/2001.09466.

[31] ZHANG Y, WALLACE B. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification[C]//Proceedings of the eighth international joint conference on natural language processing (volume 1: long papers). Taipei: Asian Federation of Natural Language Processing, 2017: 253-263.

[32] XU H, LIU B, SHU L, et al. Double embeddings and CNN-based sequence labeling for aspect extraction[C]//Proceedings of the 56th annual meeting of the Association for Computational Linguistics (volume 2: short papers). Melbourne: Association for Computational Linguistics, 2018: 592-598.

[33] MA X, HOVY E. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF[C]//Proceedings of the 54th annual meeting of the Association for

Computational Linguistics (volume 1: long papers). Berlin: Association for Computational Linguistics, 2016: 1064-1074.

[34] PENNINGTON J, SOCHER R, MANNING C. GloVe: global vectors for word representation[C]//Proceedings of the 2014 conference on empirical methods in natural language processing. Doha: Association for Computational Linguistics, 2014: 1532-1543.

[35] NGUYEN T H, GRISHMAN R. Event detection and domain adaptation with convolutional neural networks[C]//Annual meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Conference. Beijing: Association for Computational Linguistics, 2015: 365-371.

[36] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space[J]. ArXiv, 2013, abs/1301.3781.

[37] LECUN Y, BOSER B, DENKER J S, et al. Backpropagation applied to handwritten zip code recognition[J]. Neural computation, 1989, 1(4): 541-551.

[38] BOUVRIE J. Notes on convolutional neural networks[EB/OL]. [2021-11-10]. http://cogprints.org/5869/1/cnn_{tutorial}.

[39] LAFFERTY J D, MCCALLUM A, PEREIRA F C N. Conditional random fields: probabilistic models for segmenting and labeling sequence data[C]//Proceedings of the eighteenth international conference on machine learning. San Francisco: Morgan Kaufmann Publishers, 2001: 282-289.

[40] LAMPLE G, BALLESTEROS M, SUBRAMANIAN S, et al. Neural architectures for named entity recognition[C]//Proceedings of the 2016 conference of the North American Chapter of the Association for Computational Linguistics: human language technologies. San Diego: Association for Computational Linguistics, 2016: 260-270.

[41] YU F, KOLTUN V. Multi-scale context aggregation by dilated convolutions[J]. CoRR, 2016, abs/1511.07122.

[42] STRUBELL E, VERGA P, BELANGER D, et al. Fast and accurate entity recognition with iterated dilated convolutions[C]//Proceedings of the 2017 conference on empirical methods in natural language processing. Copenhagen: Association for Computational Linguistics, 2017: 2670-2680.

[43] 李妮, 关焕梅, 杨飘, 等. 基于 BERT-IDCNN-CRF 的中文命名实体识别方法 [J]. 山东大学学报 (理学版), 2020, 55(01): 102-109.

Author Contributions

Li Xuhui: Proposed research ideas, provided methodological guidance, revised the paper.
Cheng Wei: Collected, cleaned, and analyzed data; implemented the research

scheme; wrote code; drafted and revised the paper.
Tang Xiaoya: Implemented the research scheme; revised the paper.
Yu Tao: Performed data preprocessing; provided guidance on the research process; revised the paper.
Chen Zhuang: Provided guidance on the research process; proposed revision suggestions.
Qian Tieyun: Provided guidance on the research process; revised the paper.

Note: Figure translations are in progress. See original paper for figures.