Transfer Learning for Small-scale Financial Event Extraction

Xingxing Ning[1+], Shuheng Li[1+], Shijing Ding[2], Yu Nie[1*]

[1] Hanvon (Wuhan), Wuhan 430070, China
[2] CCB Fintech, 430079, China
{ningxingxing,nieyu}@hanwang.com.cn
shl060@ucsd.edu
dingshijing.wh@ccbft.com

Abstract. Event extraction is an important problem in information extraction and NLP. CCKS-2020 Shared Task on small-scale financial event extraction raises a challenging scenario on building event extraction systems. To deal with the problem of small-scale data and the requirement of adapting to new types of events, we propose to use transfer learning on the basis of document-level event extraction paradigm. Our method is composed of a BERT-based Event-classification model and a BERT-based Event-NER models for each type of event. We apply transfer learning strategy to the BERT weights of Event-NER models and identify the effectiveness via experiments. Our best ensemble model achieves F1 score of 0.85071 on testing set and ranks top 3 in the competition.

Keywords: Financial Event, Event Extraction, Named Entity Recognition, Transfer Learning.

1 Introduction

Event extraction (EE) is a classic problem in information extraction (IE) and NLP [1]. It plays a vital role in natural language processing since it can produce valuable structured information to facilitate a variety of tasks, such as knowledge base construction, question answering, language understanding, etc.

One promising direction is enhancing the finance-related research performed by finance analysts. In financial domain, large amounts of announcements call EE for assisting people in extracting valuable structured information to sense emerging risks and find profitable opportunities timely.

However, large scale fine-grained annotations on many kinds of financial event texts are required for supervised methods to work well. Due to the lack of rich annotated datasets, it is essential to apply transfer learning skills to those powerful supervised methods.

* Corresponding author: Yu Nie (nieyu@hanwang.com.cn).
+ The first two authors contributed equally to the work.

CCKS-2020 Shared Task on small-sample financial event extraction with transfer learning raises a challenging scenario on building an EE system on the financial domain. Specifically, the first challenge is that most of the financial texts contain more than one event. The other challenge is that the EE model trained on the training events set should be able to perform well when transferred to the testing events. There are limited additional training examples for testing events, which is much smaller than the training set.

In this work, we propose a transfer learning approach to tackle these two challenges. We introduce BERT weights transfer mechanism to make full use of all the training data. Experiments are conducted to test the effectiveness of our strategy and an ensemble model is finally implemented on the testing set to accomplish the CCKS-2020 Shared Task and the effectiveness of our approach has been proved accordingly.

2 Related Work

2.1 Event Extraction

Event extraction granularity divides extraction paradigms into two types: (i) document-level paradigms which assume that a piece of text refers to a single event [2], and (ii) sentence-level paradigms which assume that a single sentence describes one or more events.

Most work in event extraction has focused on the ACE sentence-level event task [3], which requires the detection of an event trigger and extraction of its arguments from within a single sentence. Previous state-of-the-art methods include Li et al. [4] and Li et al. [5], which explored a variety of hand-designed features.

Also, document-level event extraction has been explored in recent works, using hand-designed features for both local and additional context [6,7,8], and with end-to-end sequence tagging based models with contextualized pre-trained representations [9].

We use a document-level event extraction paradigm which is composed of Event-classification and Event-Name-entity-recognition (Event-NER).

2.2 Dealing with Small-scale Data

A big obstacle for democratizing EE is the lack of training data due to the enormous cost to obtain expert annotations. To tackle this problem, pre-trained language models [10,11,12] have pushed performance in many natural language processing tasks to new heights. In cases where the target task has limited labeled data, prior work has employed transfer learning by pre-training on a source dataset with abundant labeled data before fine-tuning on the target task dataset [13,14,15].

In this CCKS-2020 shared task, we use a pre-trained language model as our base EE model. And we remark, instead of complex data-enhancement, such that different types of financial events text data can have implicit structure which can be taken ad-

vantage of by our base EE model pre-trained on a naive NER task on all the training events data.

3 Approach

Our method is composed of an Event-classification model and an Event-NER model for each type of event. We firstly use BERT-based classification model for Event-classification. Then, combining with the result of Event-classification, we proposed to train separate BERT-CRF models for different Event-NER tasks. For different types of event, we train separate BERT-CRF models using transfer learning strategy, in order to fully utilize all the training examples and deal with the challenge of small-scale data. We will go into detail of the two part of our method and illustrate our transfer learning strategy in the following subsections.

3.1 Event-classification

We fine-tune a pre-trained BERT model for Event-classification. The input of the model is a sequence that has two special tokens [CLS] and [SEP]. [CLS] represents the start of a sequence and [SEP] represents the end of a sequence. BERT model encodes all the tokens in the input sequence using multiple Transformer blocks and self-attention heads. Our model regards the final hidden representation of the first token [CLS] as the representation of the whole sequence. Then it uses a linear layer and a softmax classifier to predict the probability of each event type.

3.2 Event-NER

We also fine-tune a pre-trained BERT model for each type of Event-NER. The input is a sequence that has been classified as this type of event and pre-processed similarly as the Event-classification model. Instead of using the representation of the first token, Event-NER model feeds the hidden representations of all the word tokens into a following CRF layer that models the transition score of this type of Event-NER. For all the tokens, CRF layer outputs a score for each event-entity and uses Viterbi algorithm to compute the optimal labelling of the whole sequence.

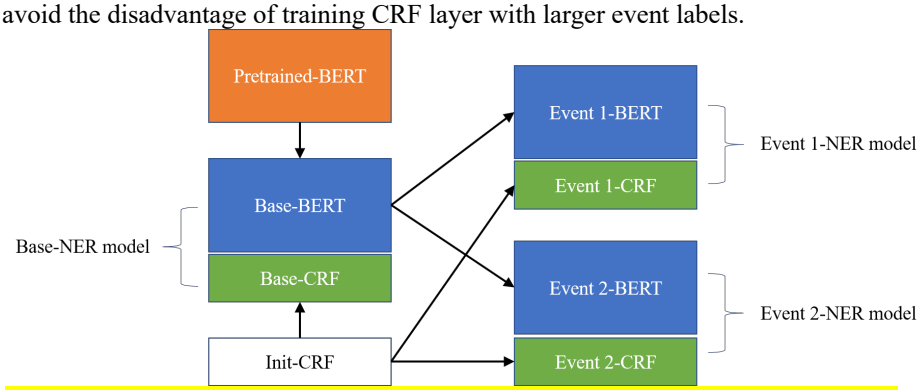
3.3 Transfer Learning for Event-NER

For each Event-NER model, it only takes the input that has been categorized as this type of event, which makes all the Event-NER models event-specific but also limits the amount of training examples. To tackle this tricky problem, we propose to firstly train a Base-NER model using all the training examples and then transfer the Base-NER model to each Event-NER model. The key insight of our method is that different types of financial events text data can have implicit structure which can be taken advantage of. Since the task of Base-NER model and Event-NER models are identical, transfer learning can also help fine-tune BERT weights to NER tasks.

Figure 1 shows how Event-NER models are transferred from the Base-NER model in detail. The BERT weights of Event-NER models are initialized using the BERT

weights of Base-NER model. However, the CRF weights of Event-NER models are not transferred because different types of event have different paradigms, therefore, have different NER labels.

Base-NER model takes all the event NER labels of different types as the base NER labels, which enables it to accept all the training examples. More training data implies that it is helpful for further tuning BERT weights of Base-NER model. But more NER labels also make Base-CRF layer difficult to converge. On the contrary, less training examples harm the tuning of Event-BERT weights but helps train Event-CRF layers. Our transfer learning strategy takes the advantage of further tuning BERT weights using larger training data through transfer and also prevents the transfer of CRF layer to avoid the disadvantage of training CRF layer with larger event labels.



(caption: This figure shows an example of our transfer learning strategy. The arrow represents how the parameters are initialized.)

4 Evaluation

In this section, we outline the experimental setup, our baselines for the task, and the influences of our strategies applied on our baseline model. Our final model performance beats most of the teams on the CCKS-2020 Leaderboard and ranks top-3 among all the teams.

4.1 Dataset

The training dataset of CCKS-2020 Shared Task on small-scale financial event extraction consists of 2732 labeled examples within 5 types of training event and 820 labeled examples within another 5 types of testing event. The validation set contains 163763 unlabeled examples within 5 training types and 60731 unlabeled examples within the first 2 testing types. The testing set contains 32879 unlabeled examples with in the 5 testing types. Leaderboard A measures the performance on validation set, while Leaderboard B measures the performance on testing set.

4.2 Evaluation metric

The task uses F1 score to evaluate the performance of financial event extraction. The F1 score is calculated as

$$F1 = \frac{2\mathrm{PR}}{\mathrm{P} + \mathrm{R}}$$

in which P is calculated as

$$\mathrm{P} = \frac{\sum_{\mathrm{event_i}}^{\mathrm{m}} \sum_{\mathrm{entity_j}}^{k} \frac{\mathrm{x}}{\mathrm{k}}}{\mathrm{m}}$$

and R is calculated as

$$R = \frac{\sum_{event_i}^{m} \sum_{entity_j}^{k} \frac{x}{k}}{n}$$

where m denotes the total number of extracted events, n denotes the total number of ground truth events and x is a bool number that represents whether the entity is correct.

4.3 Experimental Setup

We use RoBERTa-wwm-ext-large[1] as the pretrained BERT model for both Event-classification model and Event-NER models. For Event-classification model, the max sequence length is 256, the batch size is 8 and the learning rate is 2e-5. For all the Event-NER models, the max sequence length is 400, the batch size is 8 and the learning rate is 5e-5 with learning rate decay strategy applied.

We also ensemble Event-NER model to further improve the performance of our model. Specifically, we train 10 Event-NER models with different random seeds for each type of event and merge the output of the 10 models via voting. Since we use document-level event extraction paradigm for this task, the extraction of trigger entity is error-prone. Therefore, we statistically post-process the output of Event-NER models to revise the trigger entity.

4.4 Result

We compare the performance of three versions of our method, Base model, Base-transfer model and Base-transfer-ensemble model. Particularly, Base model is composed of the Event-classification model and the Event-NER models for each type of event without transfer learning. Base-transfer model applies transfer learning and Base-transfer-ensemble model applies transfer learning as well as model ensemble. The three model save the best weights through testing on a hold-out set, which is separated from the training examples.

Table

	Validation Set (F1)	Testing Set (F1)
Base	0.67215	0.81798
Base-transfer	0.69007	0.84354

[1] https://github.com/ymcui/Chinese-BERT-wwm

Base-transfer-ensemble	--	0.84906

As shown in Table 1, Base-transfer model outperforms Base model significantly on both sets, which proves the effectiveness of our transfer learning approach. Although we did not implement model ensemble on validation set, the performance of Base-transfer-ensemble on testing set also shows its effect. The best result on validation set is 0.69406 and the best result on testing set is 0.85071. The best model uses all the training examples for each type of event to train Event-NER models and save the weights after 15 epochs, instead of saving the best weights via a hold-out set. The best results rank top 3 in both leaderboards.

5 Conclusion and Discussion

Our final system of the CCKS-2020 Shared Task on small-sample financial event extraction with transfer learning is a novel solution dealing with the multi-events and small-scale challenging scenarios. We present a simple and easy-to-implement transfer learning strategy to improve the performance of the EE model on financial text. We conduct experiments to identify the effect of the strategy and implement an ensemble model on testing set to further improve the performance for the competition. The final submission result of our final model on testing set ranks top-3 among all the team submissions on the CCKS-2020 Leader board.

The key insight of our transfer learning strategy is that different types of financial events data can have similar implicit structures. Transfer learning would help BERT weights adapt to new tasks and new type of events. Following this, in the future, we will focus on further pre-training domain-specific BERT weights for different domain and tasks.

References

1. Daniel Jurafsky and James H Martin. 2009. Speech and language processing.
2. Ralph Grishman and Beth Sundheim. 1996. Message understanding conference- 6: A brief history. In COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics.
3. Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. Linguistic Data Consortium, Philadelphia, 57.
4. Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 73–82, Sofia, Bulgaria. Association for Computational Linguistics.
5. Xiang Li, Thien Huu Nguyen, Kai Cao, and Ralph Grishman. 2015. Improving event detection with abstract meaning representation. In Proceedings of the First Workshop on Computing News Storylines, pages 11–15, Beijing, China. Association for Computational Linguistics.
6. Siddharth Patwardhan and Ellen Riloff. 2009. A unified model of phrasal and sentential evidence for information extraction. In Proceedings of the 2009 Conference on Empirical

Methods in Natural Language Processing, pages 151–160, Singapore. Association for Computational Linguistics.
7. Ruihong Huang and Ellen Riloff. 2011. Peeling back the layers: Detecting event role fillers in secondary contexts. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 1137–1147, Portland, Oregon, USA. Association for Computational Linguistics.
8. Ruihong Huang and Ellen Riloff. 2012. Modeling textual cohesion for event extraction. In Twenty-Sixth AAAI Conference on Artificial Intelligence.
9. Xinya Du and Claire Cardie. 2020. Document-level event role filler extraction using multi-granularity contextualized encoding. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8010–8020, Online. Association for Computational Linguistics.
10. Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. OpenAI Blog, 1(8):9.
11. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
12. Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. arXiv 1907.11692.
13. Sewon Min, Minjoon Seo, and Hannaneh Hajishirzi. 2017. Question answering through transfer learning from large fine-grained supervision data. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 510–517, Vancouver, Canada. Association for Computational Linguistics.
14. Yu-An Chung, Hung-Yi Lee, and James Glass. 2018. Supervised and unsupervised transfer learning for question answering. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1585–1594, New Orleans, Louisiana. Association for Computational Linguistics.
15. Georg Wiese, Dirk Weissenborn, and Mariana Neves. 2017. Neural domain adaptation for biomedical question answering. In Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), pages 281–289, Vancouver, Canada. Association for Computational Linguistics.
16.