Event Detection with Entity Markers

Emanuela Boros[1][0000−0001−6299−9452],
Jose G. Moreno[1,2][0000−0002−8852−5797], and
Antoine Doucet[1][0000−0001−6160−3356]

[1] University of La Rochelle, L3i, F-17000, La Rochelle, France
emanuela.boros,antoine.doucet@univ-lr.fr
[2] University of Toulouse, IRIT, UMR 5505 CNRS, F-31000, Toulouse, France
jose.moreno@irit.fr

Abstract. Event detection involves the identification of instances of specified types of events in text and their classification into event types. In this paper, we approach the event detection task as a relation extraction task. In this context, we assume that the clues brought by the entities participating in an event are important and could improve the performance of event detection. Therefore, we propose to exploit entity information explicitly for detecting the event triggers by marking them at different levels while fine-tuning a pre-trained language model. The experimental results prove that our approach obtains state-of-the-art results on the ACE 2005 dataset.

Keywords: Information Extraction · Event Extraction · Event Detection

1 Introduction

Event detection (ED) aims to identify the instances of specified types of events in text. An event is represented by an *event mention* (a text that contains an event of a specific type and subtype), an *event trigger* (the word that expresses the event mention), an *event argument* (a participant in the event of a specific type), and an *argument role* (the role of the entity in the event). For instance, according to the ACE 2005 annotation guidelines[3], in the sentence "***She****'s been* *convicted of* ***obstruction of justice****.*", an event detection system should be able to recognize the word *convicted* as a trigger for the specific event type **Convict**.

A main challenge intervenes when the same event might appear in the form of various trigger expressions and an expression might represent different event types in different contexts. For example, *transfer* could refer to transferring ownership of an item, transferring money, or transferring personnel from one location to another. Each sense of the word is linked with an event type. In the same manner, *fired* can correspond to an **attack** type of event as in "*an American tank fired on the street*" or it can express the **dismissal** of an employee from

[3] https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-events-guidelines-v5.4.3.pdf

a job as in "*Hillary Clinton was fired from the House Judiciary Committee's Watergate investigation*".

Therefore, we would assume that, in such cases, significant clues can be given by the context of a candidate trigger and by the presence of the participants at the event in this context, e.g. named entities. For analyzing the importance of these indicators of the existence of an event in a sentence, we adopt a relation extraction model to perform event detection by taking advantage of the participants in the event (event arguments).

2 Related Work

Most current state-of-the-art systems perform event detection individually [2, 17, 6], where the entities are either ignored or considered helpful in joint models.

Some works made use of gold-standard entities in different manners. Higher results can be obtained with gold-standard entity types [17], by concatenating randomly initialized embeddings for the entity types. A graph neural network (GNN) based on dependency trees [18] has also been proposed to perform event detection with a pooling method that relies on entity mentions aggregating the convolution vectors. Arguments provided significant clues to this task in the supervised attention mechanism proposed to exploit argument information explicitly for event detection [11], while also using events from FrameNet.

Although some joint learning-based methods have been proposed, which tackled event detection and argument extraction simultaneously, these approaches usually only make significant improvements on the argument extraction, but insignificant to event detection. These methods usually combine the loss functions of these two tasks and are jointly trained under the supervision of annotated triggers and arguments. Event triggers and their arguments are predicted at the same time in a joint framework [15] with bidirectional recurrent neural networks (Bi-RNNs) and a convolutional neural network (CNN) and systematically investigate the usage of memory vectors/matrices to store the prediction information during the course of labeling sentence features.

The architecture adopted in [12] was to jointly extract multiple event triggers and event arguments by introducing syntactic shortcut arcs derived from the dependency parsing trees to enhance the information flow in an attention-based graph convolution network (GCN) model. The gold-standard entity types are embedded as features for trigger and argument prediction. The argument information was also exploited in [11] explicitly for event detection by experimenting with different strategies for adding supervised attention mechanisms. The authors exploit the annotated entity information by concatenating the token embeddings with randomly initialized entity type embeddings.

Recently, different approaches that include external resources and features at a sub-word representation level have been proposed. Thus, generative adversarial networks (GANs) have been applied in event detection [24, 8]. Besides, reinforcement learning (RL) is used in [24] for creating an end-to-end entity and event extraction framework. The approach attempted in [23] based on the BERT model with an automatic generation of labeled data by editing prototypes and filtering out the labeled samples through argument replacement by ranking their quality. A similar framework is proposed by [22] but information is encoded by BERT or a CNN suggesting a growing interest in adversarial models. Simultaneously, an integration of a distillation technique to enhance the adversarial prediction was explored in [13].

Although recent advances are focused on multiple techniques, several BERT-based architectures have been proposed [21, 23, 22]. In this work, we demonstrate that the advantages of BERT can be improved by adding extra information by explicitly marking the entities in the input text. We continue with the presentation of our proposed model in Section 3. The experimental setup and the results are detailed in Section 4 and we finalize with some conclusions and perspectives in Section 5.

3 Approach

We implemented the BERT-based model with *EntityMarkers*[4] We adapt the method presented in [19] applied for relation classification, to perform event detection. First, our model extends the BERT [3] model applied to sequential data. BERT itself is a stack of Transformer layers [20]. We refer the readers to the original paper for a more detailed description. We modify BERT by adding a conditional random fields (CRF) layer instead of the dense one, which is commonly used in other works on sequential labeling [9, 14] to ensure output consistency. Next, the *EntityMarkers* model [19] consists in augmenting the input data with a series of special tokens. Thus, if we consider a sentence $x = [x_0, x_1, \ldots, x_n]$ with n tokens, we augment x with two reserved word pieces to mark the beginning and the end of each event argument mention in the sentence.

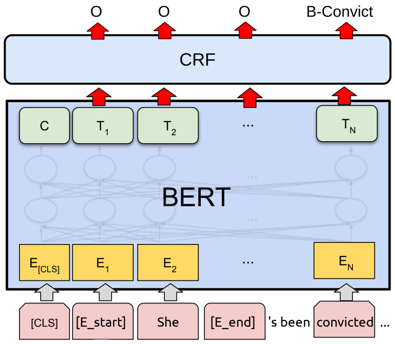

Fig. 1. The BERT-based model with *Entity Position Markers* and a CRF top layer.

[4] We only used the input type representation and consider a complex output based on tokens which is not considered in [19].

In the ACE 2005 dataset, an event argument is defined as an entity mention, a temporal expression or a value (e.g. *Crime*, *Sentence*, *Job-Title*) that is involved in an event (as participants or attributes with a specific role in an event mention). An event argument has an entity type and a role. For example, in a *Conflict.Attack* event type, one event argument can be an *Attacker* with three possible types: PER, ORG, GPE (Person, Organization, Geo-political Entity). Thus, we introduce three types of markers: (1) *Entity Position Markers*, e.g. $[E_{start}]$ and $[E_{end}]$ where E represents an entity of any type, (2) *Entity Type Markers*, e.g. PER_{start} and PER_{end} where PER represents an entity of type Person, and (3) we also test that, in the case of the event argument roles are known beforehand, the *Argument Role Markers*, e.g. $[Defendant_{start}]$, $[Defendant_{end}]$ where Defendant is an event argument role. We modify x to give:

$x = [x_0, x_1, \ldots, [MARKER_{start}]x_i \ldots x_{j-1}[MARKER_{end}], \ldots, x_n]$ and we feed this token sequence into BERT instead of x. We also update the entity indices $E = (i+1, j+1)$ to account for the inserted tokens, as shown in Figure 1 for the model with *Entity Position Markers*.

As an example, in the sentence "***She****'s been convicted of* ***obstruction of justice****.*", where *She* has the argument role of a **Defendant** and *obstruction of justice* is an argument of type **Crime**, the sentence is augmented as follows:
(1) $[E_{start}]$ ***She*** $[E_{end}]$ *'s been convicted of* $[E_{start}]$ ***obstruction of justice***$[E_{end}]$.
(2) $[PER_{start}]$ ***She*** $[PER_{end}]$ *'s been convicted of* $[Crime_{start}]$ ***obstruction of justice*** $[Crime_{end}]$.
(3) $[Defendant_{start}]$ ***She*** $[Defendant_{end}]$ *'s been convicted of* $[Crime_{start}]$ ***obstruction of justice*** $[Crime_{end}]$.

For the *Argument Role Markers*, if an entity has different roles in different events that are present in the same sentence, we mark the entity with all the argument roles that it has.

4 Experiments and Results

The evaluation is conducted on the annotated data ACE 2005 corpus. For comparison purposes, we use the same test set with 40 news articles (672 sentences), the same development set with 30 other documents (863 sentences), and the same training set with the remaining 529 documents (14,849 sentences) as in previous studies of this dataset [17, 15].

Table 1. Evaluation of the BERT-based models on the blind test data.

Models	**Precision**	**Recall**	**F1**
BERT-base-uncased	71.6	68.4	70.0
BERT-base-cased	71.3	72.0	71.6
BERT-large-uncased	72.0	72.9	72.5
BERT-large-cased	69.3	77.1	73.0

The ACE 2005 corpus has 8 types of events, with 33 subtypes (e.g. the event type *Conflict* has two subtypes *Attack, Demonstrate*) that, along with one

Table 2. Evaluation of our models and comparison with state-of-the-art systems for event detection on the blind test data. $^+$with gold-standard arguments. Change improvements w.r.t. our models are showed in columns "F1 Improvement (%)". Improvements greater than 10% are highlighted with background color. Statistical significance is measured with McNemar's test. * denotes a significant improvement at $p \leq 0.01$.

				F1 Improvement (%)		
Models	**Precision**	**Recall**	**F1**	**(1)**	**(2)**	**(3)**
CNN [17]	71.9	63.8	67.6	**12.72%**	**16.12%**	**17.75%**
CNN$^+$[17]	71.8	66.4	69.0	10.43%	13.77%	15.36%
Dynamic multi-pooling CNN [2]	75.6	63.6	69.1	10.27%	13.60%	15.20%
Joint RNN [15]	66.0	73.0	69.3	9.96%	13.28%	14.86%
CNN with document context [6]	77.2	64.9	70.5	8.09%	11.35%	12.91%
Non-Consecutive CNN [16]	N/A	N/A	71.3	6.87%	10.10%	11.64%
Attention-based$^+$ [11]	78.0	66.3	71.7	6.28%	9.48%	11.02%
GAIL [24]	74.8	69.4	72.0	5.83%	9.03%	10.56%
Gated Cross-Lingual Attention [10]	78.9	66.9	72.4	5.25%	8.43%	9.94%
Graph CNN [18]	77.9	68.8	73.1	4.24%	7.39%	8.89%
Seed-based [1]	80.6	67.1	73.2	4.10%	7.24%	8.74%
Hybrid NN [7]	**84.6**	64.9	73.4	3.81%	6.95%	8.45%
Attention-based GCN [12]	76.3	71.3	73.7	3.39%	6.51%	8.01%
Δ-learning [13]	76.3	71.9	74.0	2.97%	6.08%	7.57%
DEEB-RNN3y [25]	72.3	75.8	74.0	2.97%	6.08%	7.57%
BERT-base-uncased+LSTM [21]	N/A	N/A	68.9	**10.60%**	**13.93%**	**15.53%**
BERT-base-uncased [21]	N/A	N/A	69.7	9.33%	12.63%	14.20%
BERT-base-uncased [5]	67.1	73.2	70.0	8.86%	12.14%	13.71%
BERT-QA [5]	71.1	73.7	72.3	5.39%	8.58%	10.10%
DMBERT [22]	77.6	71.8	74.6	2.14%	5.23%	6.70%
DMBERT+Boot [22]	77.9	72.5	75.1	1.46%	4.53%	5.99%
BERT-large-cased	69.3	77.1	73.0	4.38%	7.53%	9.04%
BERT-large-cased+*Entity Position Markers*$^+$ (1)	75.9	76.6	76.2*	-	3.02%	4.46%
BERT-large-cased+*Entity Type Markers*$^+$ (2)	79.3	77.8	78.5*	-	-	1.40%
BERT-large-cased+*Argument Role Markers*$^+$ (3)	78.9	**80.4**	**79.6***	-	-	-

class "O" for the non-trigger tokens, constitutes a 34-class classification problem. Following the same line of works, we consider that a trigger is correct if its event type, subtype, and offsets match those of a reference trigger. We use precision (P), recall (R), and F-measure (F1) to evaluate the overall performance.

We first consider four baselines based on the BERT language model, applied in a similar way to [4] for the named entity recognition (NER) task, with the recommended hyperparameters. We test four widely used pre-trained English language models, two based on BERT-base and two based on BERT-large, *cased* (trained on the original words) and *uncased* (trained on lowercased words). tion is connected to the recognition of named entities, that are usually considered important to detect event mentions.

We compare our proposed models with markers with several state-of-the-art neural-based models proposed for event detection, that do not use external resources, more specifically with the following models based on CNNs and RNNs: the CNN-based model [17] with and without the addition of gold-standard entities, the dynamic multi-pooling CNN model [2], the bidirectional joint RNNs [15], the non-consecutive CNN in [16], the hybrid model [7], the GAIL model [24], the gated cross-lingual attention model [10], and the graph CNN [18].We also compare our approach with recent proposed BERT-based models, the finetuned baseline BERT-base-uncased [5], the QA-BERT [5] where the task has been approached as a question answering task, the two models with adversarial training for weakly supervised event detection [22], and the BERT and LSTMs approaches [21] that models text spans and captures within-sentence and cross-sentence context.

Between the BERT-based baseline models presented in Table 1, it is worth noticing that the *cased* models perform better than the *uncased* ones, which could confirm that named entities that are usually capitalized are an important clue for the event detection task[5]. Moreover, the results are similar to the BERT-base-uncased in [5] (the same F1 value and similar precision and recall scores) and [21].

Full results of our model and its comparison against state of the art is presented in Table 2. There is a significant gain with the trigger classification of 9.04% higher over the stand-alone BERT-based model and 5.99% to the best reported previous models. These results demonstrate the effectiveness of our method to incorporate the argument information.

Moreover, the improvements are consistent regardless of the type of encoder (BERT or other) used to represent the inputs. For our first model (*Entity Position Markers*), where the entities are surrounded by a general marker that does not depend on the entity type, the results are improved with three percentage points revealing that the position of the entities is relevant for the trigger detection task. Furthermore, when we mark the entities with their argument roles (*Argument Role Markers*), the recall and F1 increase with around one absolute percentage point. However, this case is substantially optimistic as it assumes that argument roles were correctly identified and typed.

5 Conclusions and Perspectives

We presented an approach for integrating entity information for the event detection task by adding different levels of entity markers, their positions, their types, and finally, their argument roles. Considering the results, we can conclude that marking entities in a sentence can significantly improve the F1 scores and obtain state-of-the-art values. Further analysis remains to be done in order to understand in which cases the markers bring informative features. As future work, we propose to tackle the drawbacks of our current model by introducing the recognition and typing of the entities in our model.

Acknowledgments

This work has been supported by the European Union's Horizon 2020 research and innovation program under grants 770299 (NewsEye) and 825153 (Embeddia).

[5] An amount of around 30% of the entities and 3% of the event triggers have the first token capitalized.

References

1. Bronstein, O., Dagan, I., Li, Q., Ji, H., Frank, A.: Seed-based event trigger labeling: How far can event descriptions get us? In: ACL (2). pp. 372–376 (2015)
2. Chen, Y., Xu, L., Liu, K., Zeng, D., Zhao, J.: Event extraction via dynamic multi-pooling convolutional neural networks. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. vol. 1, pp. 167–176 (2015)
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019). pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019)
5. Du, X., Cardie, C.: Event extraction by answering (almost) natural questions. arXiv preprint arXiv:2004.13625 (2020)
6. Duan, S., He, R., Zhao, W.: Exploiting document level information to improve event detection via recurrent neural networks. In: Eighth International Joint Conference on Natural Language Processing (IJCNLP 2017). pp. 352–361. Asian Federation of Natural Language Processing (2017)
7. Feng, X., Huang, L., Tang, D., Ji, H., Qin, B., Liu, T.: A language-independent neural network for event detection. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). vol. 2, pp. 66–71 (2016)
8. Hong, Y., Zhou, W., Zhang, J., Zhou, G., Zhu, Q.: Self-regulation: Employing a generative adversarial network to improve event detection. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 515–526 (2018)
9. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360 (2016)
10. Liu, J., Chen, Y., Liu, K., Zhao, J.: Event detection via gated multilingual attention mechanism. In: Thirty-second AAAI Conference on Artificial Intelligence (AAAI-18) (2018)
11. Liu, S., Chen, Y., Liu, K., Zhao, J.: Exploiting argument information to improve event detection via supervised attention mechanisms. In: 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017). pp. 1789–1798. Vancouver, Canada (2017)
12. Liu, X., Luo, Z., Huang, H.: Jointly multiple events extraction via attention-based graph information aggregation. arXiv preprint arXiv:1809.09078 (2018)
13. Lu, Y., Lin, H., Han, X., Sun, L.: Distilling discrimination and generalization knowledge for event detection via delta-representation learning. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 4366–4376 (2019)
14. Ma, X., Hovy, E.: End-to-end sequence labeling via bi-directional lstm-cnns-crf. arXiv preprint arXiv:1603.01354 (2016)
15. Nguyen, T.H., Cho, K., Grishman, R.: Joint event extraction via recurrent neural networks. In: Proceedings of NAACL-HLT. pp. 300–309 (2016)

16. Nguyen, T.H., Fu, L., Cho, K., Grishman, R.: A two-stage approach for extending event detection to new types via neural networks. ACL 2016 p. 158 (2016)
17. Nguyen, T.H., Grishman, R.: Event detection and domain adaptation with convolutional neural networks. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. pp. 365–371 (2015)
18. Nguyen, T.H., Grishman, R.: Graph convolutional networks with argument-aware pooling for event detection. In: Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018) (2018)
19. Soares, L.B., FitzGerald, N., Ling, J., Kwiatkowski, T.: Matching the blanks: Distributional similarity for relation learning. arXiv preprint arXiv:1906.03158 (2019)
20. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
21. Wadden, D., Wennberg, U., Luan, Y., Hajishirzi, H.: Entity, relation, and event extraction with contextualized span representations. arXiv preprint arXiv:1909.03546 (2019)
22. Wang, X., Han, X., Liu, Z., Sun, M., Li, P.: Adversarial training for weakly supervised event detection. 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). pp. 998–1008 (2019)
23. Yang, S., Feng, D., Qiao, L., Kan, Z., Li, D.: Exploring pre-trained language models for event extraction and generation. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 5284–5294 (2019)
24. Zhang, T., Ji, H., Sil, A.: Joint entity and event extraction with generative adversarial imitation learning. Data Intelligence **1**(2), 99–120 (2019)
25. Zhao, Y., Jin, X., Wang, Y., Cheng, X.: Document embedding enhanced event detection with hierarchical and supervised attention. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 414–419 (2018)