Word-Label Alignment for Event Detection: A New Perspective via Optimal Transport

Amir Pouran Ben Veyseh
Department of Computer and Information Science
University of Oregon
Eugene, Oregon, USA
apouranb@cs.uoregon.edu

Thien Huu Nguyen
Department of Computer and Information Science
University of Oregon
Eugene, Oregon, USA
thien@cs.uoregon.edu

Abstract

Event Detection (ED) aims to identify mentions/triggers of real world events in text. In the literature, this task is modeled as a sequence-labeling or word-prediction problem. In this work, we present a novel formulation in which ED is modeled as a word-label alignment task. In particular, given the words in a sentence and possible event types, the objective is to infer an alignment matrix in which event trigger words are aligned with the most likely event types. Moreover, we show that this new perspective facilitates the incorporation of word-label alignment biases to improve alignment matrix for ED. Novel alignment biases and Optimal Transport are introduced to solve our alignment problem for ED. We conduct experiments on a benchmark dataset to demonstrate the effectiveness of the proposed model for ED.

1 Introduction

Event Detection (ED) is one of the critical tasks in Information Extraction. Its goal is to identify and classify event triggers, i.e., the words/phrases that most clearly refer to the occurrence of an event of some predefined types in text. For example, in the sentence "*Joe Biden was born on November 20, 1942*", an ED system should recognize the word "*born*" as a trigger word of an event of type *Birth*.

A major challenge for ED is to assign an appropriate event type label for each word in a given sentence. In this work, we introduce a new perspective to solve ED as a word-label alignment problem that aims to align the set of words in the input sentence with the set of possible event type labels to represent correct label assignment for words. A key requirement for ED models in this new perspective involve inferring an alignment matrix to capture an alignment likelihood score between each pair of words and label types. The models can then be trained by enforcing the similarity between the predicted alignment matrix and the golden alignment matrix (computed from training data). In this way, previous ED models can be seen as a way to achieve the alignment matrix between words and labels where label distributions computed by the models serve as the alignment likelihood scores (Nguyen and Grishman, 2015; Chen et al., 2015; Wang et al., 2019; Cui et al., 2020; Ngo et al., 2021). However, given the word-label alignment perspective, previous ED models are suboptimal in at least two ways. First, the alignment likelihood scores in prior models are only used locally for each word (i.e., to compute the cross-entropy loss for each word to train models). The global uses of alignment matrix (e.g., to compute an overall distance between words and labels for training signals) are thus not yet explored in previous ED models. Second, current ED models mainly obtain alignment likelihood scores based on representation vectors for words and types, thus unable to exploit assignment biases to improve quality of the alignment matrix to train ED models. In particular, we propose two types of alignment biases that can be helpful for ED: (1) Word Preference: words with high likelihoods to be event triggers should be more aligned with event type labels (i.e., not the *Other* type for non-trigger words), and (2) Type Preference: event types that have higher chance to be appear in the input sentence should be associated with greater alignment scores. In all, we expect that global application and alignment biases can provide complementary information to boost current ED models in the new perspective.

To implement this idea, we propose to encode event trigger likelihoods for words and appearance likelihoods for event types as two distributions over words and event type labels (respectively) that will be induced from a deep learning architecture. Next, to inject the alignment biases into our ED model, we propose to feed the two distributions into Optimal Transport (OT) (Peyre and Cuturi, 2019) to induce an alignment matrix between words and event type labels. OT is an established framework

Proceedings of the 11th Joint Conference on Lexical and Computational Semantics, pages 132 - 138

to find the optimal alignment between two distributions, thus providing a decent solution to incorporate alignment biases to compute alignment matrix in our ED problem. Finally, the induced alignment matrix will be leveraged to obtain a distance between words and event type labels, serving as a global application of the alignment matrix to introduce new training signals for ED. We conduct extensive experiments on a benchmark dataset to deliver state-of-the-art performance for ED. In summary, our contributions include:

- A new perspective based on word-label alignment for event detection.
- Introduction of optimal transport to incorporate novel alignment biases for event detection.
- State-of-the-art performance for sequence-labeling event detection.

2 Model

Given an input sentence $S = [w_1, w_2, \ldots, w_n]$, the goal of ED is to predict the label sequence $L = [l_1, l_2, \ldots, l_n]$ where $l_i \in \mathcal{T}$ is the label for the word $w_i \in S$. Here, the label set $\mathcal{T}$ involves the BIO encoding tags for the event types in a given event ontology (e.g., *B_Birth*, *I_Birth*, and *Other*). In this work, we propose to model ED as a word-label alignment problem where an alignment matrix is formed to capture the assignment likelihood for every pair of words in S and labels in $\mathcal{T}$. We will first discuss word/label representations, and alignment matrix computation for training afterward.

Word & Label Representation: To represent the words in S, following prior work (Wang et al., 2019), we employ the pre-trained BERT model (Devlin et al., 2019). Concretely, the input sentence $[[CLS], w_1, w_2, \ldots, w_n]$ is fed into BERT to compute the contextualized embedding vectors $E = [e_{cls}, e_1, e_2, \ldots, e_n]$. We employ the average of vectors in the last layer of BERT to produce E. For the words with multiple word-pieces, we take the average of their word-piece representations.

To represent the event type labels l_i, we employ a randomly initialized embedding table T in which every label is represented by a vector t_i. The representations of the labels are updated during training.

Alignment: To predict the label sequence L with our alignment idea, for every word w_i, an alignment likelihood score $a_{i,j}$ between w_i and each label l_j is required (i.e., forming an alignment matrix A). Using the scores $a_{i,j}$, the label $\bar{l}_i$ can be predicted by $\bar{l}_i = \text{argmax}_j a_{i,j}$. Note that in prior ED models, the alignment scores $a_{i,j}$ are directly computed using the final task-specific feed-forward networks (Wang et al., 2019; Veyseh et al., 2021b). This approach is equivalent to computing the similarity between the representation vectors w_i and t_j, e.g., via dot-product. We call this approach "*Vanilla Alignment*". However, as discussed in the introduction, vanilla alignment scores $a_{i,j}$ are solely dependent on the learned representations e_i and t_j. As such, they cannot incorporate the alignment biases into the alignment matrix for ED.

To this end, we introduce two alignment biases that can be exploited to improve the word-label alignment for ED. In particular, for an effective ED model, we expect the words that are more likely to be event triggers to have higher alignment scores with event types. In contrast, the other words should be better aligned with the special label *Other*. i.e., non-trigger. We call this bias "*Word Preference*" for ED. In addition, among all event types, it is expected that the event types that have higher chance to be mentioned in the input sentence to be associated with greater scores in the alignment matrix A. We name this bias as "*Type Preference*". In this work, we aim to modify the vanilla alignment approach such that the two aforementioned preferences are observed. The quantification of Word and Type Preference and their incorporation into alignment matrix will be discussed in the following.

Word & Type Preference: To compute the word preference and type preference in the input sentence S, we consider two simpler versions of the ED problem. Specifically, for word preference, we utilize the Trigger Identification (TI) task that seeks to recognize the event trigger words without classifying them by event types. The event trigger probability computed for TI can be used to quantify the event trigger likelihood for each word $w_i \in S$. Concretely, the representation e_i of w_i is fed into a feed-forward network with sigmoid activation function to compute the trigger likelihood p_i^w for w_i: $p_i^w = \sigma(FF_w(e_i))$, where σ and FF_w are sigmoid and feed-forward layer, respectively. To supervise the trigger likelihood scores, we include the binary cross-entropy loss function for TI into the overall loss for training: $\mathcal{L}_{TI} = -\frac{1}{n}\sum_{i=1}^{n}(y_i^w * \log(p_i^w) + (1 - y_i^w) * \log(1 - p_i^w))$,

where y_i^w is a binary number to indicate whether if w_i is a trigger in S. The likelihood scores p_i^w are employed to represent the word preference.

Next, for the type preference, we exploit the task of Type Prediction (TP) for ED. In this task, the objective is to predict which event types are mentioned in the sentence S (i.e., without predicting the trigger words). For an event type label t_j, we predict the likelihood for t_j to be mentioned in S by concatenating the type representation t_j with the sentence representation e_{cls} and feeding the result into a separate feed-forward network FF_t with sigmoid activation to obtain the appearance likelihood for t_j: $p_j^t = \sigma(FF_t([t_j, e_{cls}]))$. To supervise the appearance likelihoods, the binary cross-entropy loss function for TP is employed: $\mathcal{L}_{TP} = -\frac{1}{|\mathcal{T}|}\sum_{j=1}^{|\mathcal{T}|}(y_j^t * \log(p_j^t) + (1-y_j^t) * \log(1-p_j^t))$, where y_j^t is a binary number to indicate the appearance of the event type t_j in S. The likelihood scores p_j^t are utilized to represent the type preference.

Alignment Computation: Given the word and type preference scores p_i^w and p_j^t, how can we compute an alignment matrix A between the words in S and the event type labels in $\mathcal{T}$ that can incorporate both word-label representation similarity (as in vanilla alignment) and designed preference scores for ED? Note that the preference scores can be modeled as two distributions over words and event type labels by applying a softmax function over the word and type likelihoods: $D^{WP} = softmax(p_1^w, p_2^w, \ldots, p_n^w)$ and $D^{TP} = softmax(p_1^t, p_2^t, \ldots, p_T^t)$. As such, we propose to employ Optimal Transport (OT) to elegantly combine the information to produce the alignment matrix A between S and $\mathcal{T}$ for ED.

Formally, given the probability distributions $p(x)$ and $q(y)$ over the domains $\mathcal{X}$ and $\mathcal{Y}$, and the cost/distance function $C(x,y) : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}_+$ for mapping $\mathcal{X}$ to $\mathcal{Y}$, OT finds the optimal joint alignment/distribution $\pi^*(x,y)$ with marginals $p(x)$ and $q(y)$ that converts $p(x)$ to $q(y)$ (i.e., the cheapest plan), by solving the following problem:

$$\pi^*(x,y) = \min_{\pi \in \Pi(x,y)} \sum_{\mathcal{Y}} \sum_{\mathcal{X}} \pi(x,y)C(x,y) \quad \text{s.t. } x \sim p(x) \text{ and } y \sim q(y), \tag{1}$$

Here, $\Pi(x,y)$ involves all joint distributions with marginals $p(x)$ and $q(y)$. As such, the joint distribution $\pi^*(x,y)$ is a matrix whose entry (x,y) $(x \in \mathcal{X}, y \in \mathcal{Y})$ represents the probability of transforming x to y in the optimal transport. We use the Sinkhorn algorithm to approximately solve OT (Peyre and Cuturi, 2019). Finally, given $\pi^*(x,y)$, one approach to employ its global information is to compute the cost of optimal conversion $Dist(\pi^*) = \Sigma_{x \in \mathcal{X}} \Sigma_{y \in \mathcal{Y}} \pi^*(x,y)C(x,y)$ to measure the distance between $\mathcal{X}$ and $\mathcal{Y}$ (i.e., the Wasserstein distance).

To apply OT in our model, the domains $\mathcal{X}$ and $\mathcal{Y}$ are defined as the words $w_i \in S$ and types $t_j \in \mathcal{T}$; the distributions $p(x)$ and $q(y)$ are set to the preference distributions D^{WP} and D^{TP}; and the cost function $C(w_i, t_j)$ is computed using the Euclidean distance between the representations e_i and t_j. As such, solving the OT equation leads to the optimal alignment $\pi^*(w_i, t_j)$, serving as our predicted alignment matrix (i.e., $a_{i,j} = \pi^*(w_i, t_j)$).

To train the ED model with word-label alignment, we propose two training signals obtained from the predicted alignment $\pi^*(e_i, t_j)$. First, by treating the alignment score $\pi^*(e_i, t_j)$ as the probability for w_i to be assigned with label t_j, we employ the negative log-likelihood loss to train our model: $\mathcal{L}_{task} = -\frac{1}{n}\sum_{i=1}^{n} \log(\pi^*(w_i, l_i))$, where l_i is the golden label for w_i in S. Second, we propose to globally enforce the similarity between the predicted alignment matrix $\pi^*(w_i, t_j)$ from OT and the golden binary alignment matrix $\pi^g(w_i, t_j)$ (i.e., $\pi^g(w_i, t_j) = 1$ if only if w_i has the golden label t_j). As such, to aggregate the information in the alignment matrices, we first compute the Wasserstein distances $Dist(\pi^*)$ and $Dist(\pi^g)$ based on the predicted and golden alignments π^* and π^g. Afterward, we seek to minimize the difference between $Dist(\pi^*)$ and $Dist(\pi^g)$ to achieve alignment matrix similarity to train our ED models, leading to the loss: $\mathcal{L}_{OT} = |Dist(\pi^*) - Dist(\pi^g)|$. Finally, the overall loss function for the entire model is $\mathcal{L} = \alpha_{task}\mathcal{L}_{task} + \alpha_{OT}\mathcal{L}_{OT} + \alpha_{TI}\mathcal{L}_{TI} + \alpha_{TP}\mathcal{L}_{TP}$.

3 Experiments

Datasets & Baselines: We evaluate the performance of the proposed model (called **OTED**) on the ACE 2005 dataset (Walker et al., 2006) that annotates 599 documents for 33 event types in English. We use the same data split and preprocessing as prior work (Wang et al., 2019; Veyseh et al., 2021b) for this dataset. The numbers of documents for the training/development/test data are 529/30/40 respectively. Following (Wang et al., 2020a; Veyseh et al., 2021b), we use the sequence-labeling setting for the ED task in ACE 2005 that adheres to the original annotation to allow event

Model	ACE		
	P	R	F1
BiLSTM	77.20	74.90	75.40
DMBERT	71.49	76.95	74.12
BERT+CRF	71.30	77.10	74.10
ED3C	80.31	76.04	78.12
OTED (ours)	79.28	79.48	**79.38**

Table 1: Model performance on the test sets. OTED is significantly better than the baselines with $p < 0.05$.

Line	Model	P	R	F1
1	OTED (full)	79.12	79.94	**79.53**
2	OTED - WP	75.14	81.39	78.14
3	OTED - TP	77.32	78.55	77.93
4	OTED - WP- TP	76.90	76.92	76.91
5	OTED - $\mathcal{L}_{task}$	75.24	77.02	76.12
6	OTED - $\mathcal{L}_{OT}$	75.92	80.28	78.04
7	OTED - $\mathcal{L}_{TI}$	78.91	75.60	77.22
8	OTED - $\mathcal{L}_{TP}$	78.21	76.05	77.12
9	Distance	76.66	78.03	77.34
10	Alignment	77.98	78.93	78.45

Table 2: Model performance on the ACE 2005 dev set.

triggers to span multiple words.

As the baselines, we compare with the typical sequence labeling models for ED, i.e., **BiLSTM**, **DMBERT** (BERT with dynamic multi-pooling), and **BERT+CRF** in (Wang et al., 2020a), and the prior state-of-the-art (SOTA) model reported for ACE 2005, i.e., **ED3C** (Veyseh et al., 2021b). For all the models, we use the same version of pre-trained BERT$_{base}$ to achieve a fair comparison. Following prior work (Wang et al., 2020b; Veyseh et al., 2021b), we use span-based precision, recall and F1 scores for correctly predicting the boundaries and types of event triggers as the performance metrics. Finally, we fine-tune the hyper-parameters for OTED using the development data of ACE 2005. In our model we use the BERT$_{base}$ model to encode data; 2 layers for all the feed-forward neural networks with 200 hidden dimensions in the layers. The trade-off parameters α_{task}, α_{OT}, α_{TI} and α_{TP} are set to 1.0, 0.01, 0.05, and 0.01 respectively. The learning rate is set to $3e$-5 for the Adam optimizer and the batch size of 8 is employed during training.

Results: The model performance is presented in Table 1. This table shows that OTED significantly outperforms the baseline models on ACE 2005. We attribute the superiority of OTED to its capability to incorporate alignment biases, i.e., word and type preference, into alignment-based ED. The better performance of OTED over ED3C is important as unlike this baseline OTED does not require additional document context or supervision from other related tasks.

Ablation Study: We conduct an ablation study for the components of OTED over the ACE 2005 development set. Table 2 presents the performance of three groups of ablated models for OTED. In the first group (lines 2-4), we exclude one or both alignment biases, i.e., WP and TP, from OTED. Concretely, to remove a preference, its corresponding distribution in the OT (i.e., D^{WP} and D^{TP}) is replaced with the uniform distribution in the OT computation for OTED. It is clear from the table that both alignment biases are beneficial for OTED as removing any of them would hurt the performance significantly. Next, the second group (lines 5-8), we exclude each loss component (i.e., $\mathcal{L}_{task}$, $\mathcal{L}_{OT}$, $\mathcal{L}_{TP}$, and $\mathcal{L}_{TI}$) from the overall loss $\mathcal{L}$ to train OTED. As can be seen, all the designed losses contribute significantly to the performance of OTED, thus testifying to their effectiveness in alignment-based ED. Also, in the third group (lines 9-10), we explore two variants of OTED to justify the design of the loss $\mathcal{L}_{OT}$ to incorporate OT into the model. In one variant (called **Distance** in line 9), instead of minimizing the difference $\mathcal{L}_{OT}$ between the Wasserstein distances based on predicted and golden alignments, we directly minimize the predicted Wasserstein distance $Dist(\pi^*)$ between words and labels. Moreover, in the **Alignment** variant in line 10, instead of employing the Wasserstein distance, we directly minimize the distance between the predicted and golden alignment $\pi^*(w_i, t_j)$ and $\pi^g(w_i, t_j)$ (i.e., evaluated by $\sum_{i,j} |\pi^*(w_i, t_j) - \pi^g(w_i, t_j)|/(n|\mathcal{T}|)$). As can be seen, both **Distance** and **Alignment** lead to inferior performance for OTED, thereby showing the effectiveness of $\mathcal{L}_{OT}$ for ED. As such, we attribute the poor performance of **Distance** to the lack of supervision from the golden alignment-based distance $\pi^g(w_i, t_j)$, and the worse performance of **Alignment** to the missing of contextual similarity (i.e., the cost $C(w_i, t_j)$) in the distance computation.

Analysis: In this section, we present a qualitative analysis to shed more light on the superiority of the proposed model OTED to the prior sequence labeling methods. Specifically, we compare our model with the **BERT+CRF** baseline by analyzing the examples in which **BERT+CRF** fails to recog-

ID	Example	BERT+CRF Prediction	OTED Prediction	Gold Event Trigger & Type
1	These are the reasons that none of these mothereffers should ever see the light of day ... they need to be all lined up and **shot**.	Trigger: "*shot*", Event Type: ***Contact:Meet***	Trigger: "*shot*", Event Type: *Justice:Execute*	Trigger: "*shot*", Event Type: *Justice:Execute*
2	Well , John, given all that you've said, we know that there's an American **retired** general **waiting** in Kuwait.	Trigger: "***waiting***", Event Type: *Personnel:End-Position*	Trigger: "*retired*", Event Type: *Personnel:End-Position*	Trigger: "*retired*", Event Type: *Personnel:End-Position*

Table 3: Case study on the development set of the ACE 2005 dataset. The golden trigger words are underlined.

nize the event types and triggers, but OTED can successfully perform the predictions. A major findings in our analysis is that OTED can exploit the introduced alignment bias (i.e., word and type preference) to avoid unlikely event triggers and types (i..e, the ones that should be obviously eliminated based on overall sentence context). This leads to correct predictions for examples that **BERT+CRF** make mistakes. Table 3 shows two examples from the development set of the ACE 2005 dataset to illustrate our findings. In the first example, the baseline can recognize the event trigger "*shot*", but fails to predict the event type. Given the context of the sentence, the predicted event type *Contact:Meet* by **BERT+CRF** should be considered as unlikely to be mentioned in the sentence. As the proposed model OTED employs type preference knowledge, it successfully avoids unlikely event types for this sentence. In addition, in the second example, the baseline incorrectly predicts a non-trigger word (i.e., "*waiting*") as a trigger. In contrast, since OTED employs word preference knowledge, it can effectively avoid unlikely event triggers.

4 Related Work

Early methods for ED employed feature engineering models (Ahn, 2006; Ji and Grishman, 2008; Liao and Grishman, 2010; Hong et al., 2011; Li et al., 2013; Miwa et al., 2014; Yang and Mitchell, 2016). Recently, deep learning was adopted as the SOTA approach for ED (Chen et al., 2015; Nguyen et al., 2016; Sha et al., 2018; Nguyen and Grishman, 2018; Yang et al., 2019; Wang et al., 2019; Lai et al., 2020; Cui et al., 2020; Tong et al., 2020; Nguyen et al., 2021). Unlike such prior work, we introduce a new word-label alignment perspective using OT for ED. Finally, some recent work has utilized OT for character/word/example alignment problems (Dou and Neubig, 2021; Xu et al., 2021; Veyseh et al., 2021a, 2022; Guzman-Nateras et al., 2022). However, none of them explores OT for word-label alignment in ED.

5 Conclusion

We present a general word-label alignment formulation for ED in which each pair of words and types is associated with an alignment score for label assignment likelihood. Moreover, we introduce two alignment biases based on type and word preference to improve the word-label alignment matrix computation with OT. Extensive analysis on a benchmark dataset demonstrates the benefits of the proposed technique for ED. In the future, we plan to evaluate our method on more datasets for ED (Wang et al., 2020a; Man et al., 2020; Lai et al., 2021) to better understand its operation.

Acknowledgement

This research has been supported by the Army Research Office (ARO) grant W911NF-21-1-0112 and the NSF grant CNS-1747798 to the IU-CRC Center for Big Learning. This research is also based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA Contract No. 2019-19051600006 under the Better Extraction from Text Towards Enhanced Retrieval (BETTER) Program. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ARO, ODNI, IARPA, the Department of Defense, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein. This document does not contain technology or technical data controlled under either the U.S. International Traffic in Arms Regulations or the U.S. Export Administration Regulations.

References

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Shiyao Cui, Bowen Yu, Tingwen Liu, Zhenyu Zhang, Xuebin Wang, and Jinqiao Shi. 2020. Edge-enhanced graph convolution networks for event detection with syntactic relation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2329–2339, Online. Association for Computational Linguistics.

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 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Luis Fernando Guzman-Nateras, Minh Van Nguyen, and Thien Huu Nguyen. 2022. Cross-lingual event detection via optimized adversarial training. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Yu Hong, Jianfeng Zhang, Bin Ma, Jianmin Yao, Guodong Zhou, and Qiaoming Zhu. 2011. Using cross-entity inference to improve event extraction. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Viet Lai, Minh Van Nguyen, Heidi Kaufman, and Thien Huu Nguyen. 2021. Event extraction from historical texts: A new dataset for black rebellions. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2390–2400, Online. Association for Computational Linguistics.

Viet Dac Lai, Tuan Ngo Nguyen, and Thien Huu Nguyen. 2020. Event detection: Gate diversity and syntactic importance scores for graph convolution neural networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Shasha Liao and Ralph Grishman. 2010. Filtered ranking for bootstrapping in event extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*.

Duc Trong Hieu Man, Duc Trong Le, Amir Pouran Ben Veyseh, Thuat Nguyen, and Thien Huu Nguyen. 2020. Introducing a new dataset for event detection in cybersecurity texts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Makoto Miwa, Paul Thompson, Ioannis Korkontzelos, and Sophia Ananiadou. 2014. Comparable study of event extraction in newswire and biomedical domains. In *Proceedings of the International Conference on Computational Linguistics (COLING)*.

Nghia Trung Ngo, Duy Phung, and Thien Huu Nguyen. 2021. Unsupervised domain adaptation for event detection using domain-specific adapters. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4015–4025, Online. Association for Computational Linguistics.

Minh Van Nguyen, Viet Dac Lai, and Thien Huu Nguyen. 2021. Cross-task instance representation interactions and label dependencies for joint information extraction with graph convolutional networks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Thien Huu Nguyen and Ralph Grishman. 2018. Graph convolutional networks with argument-aware pooling for event detection. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*.

Gabriel Peyre and Marco Cuturi. 2019. Computational optimal transport: With applications to data science. In *Foundations and Trends in Machine Learning*.

Lei Sha, Feng Qian, Baobao Chang, and Zhifang Sui. 2018. Jointly extracting event triggers and arguments by dependency-bridge rnn and tensor-based argument interaction. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*.

Meihan Tong, Bin Xu, Shuai Wang, Yixin Cao, Lei Hou, Juanzi Li, and Jun Xie. 2020. Improving event detection via open-domain trigger knowledge. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Amir Pouran Ben Veyseh, Viet Lai, Franck Dernoncourt, and Thien Huu Nguyen. 2021a. Unleash GPT-2 power for event detection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Amir Pouran Ben Veyseh, Minh Van Nguyen, Franck Dernoncourt, Bonan Min, and Thien Huu Nguyen. 2022. Document-level event argument extraction via optimal transport. In *Findings of the Association for Computational Linguistics: ACL 2022*.

Amir Pouran Ben Veyseh, Minh Van Nguyen, Nghia Ngo Trung, Bonan Min, and Thien Huu Nguyen. 2021b. Modeling document-level context for event detection via important context selection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5403–5413, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. In *Technical report, Linguistic Data Consortium*.

Xiaozhi Wang, Xu Han, Zhiyuan Liu, Maosong Sun, and Peng Li. 2019. 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)*, pages 998–1008.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020a. MAVEN: A Massive General Domain Event Detection Dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020b. Maven: A massive general domain event detection dataset. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Jingjing Xu, Hao Zhou, Chun Gan, Zaixiang Zheng, and Lei Li. 2021. Vocabulary learning via optimal transport for neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Bishan Yang and Tom M. Mitchell. 2016. Joint extraction of events and entities within a document context. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Exploring pre-trained language models for event extraction and generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5284–5294, Florence, Italy. Association for Computational Linguistics.