Leveraging FrameNet to Improve Automatic Event Detection

Shulin Liu, Yubo Chen, Shizhu He, Kang Liu and Jun Zhao
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
{shulin.liu, yubo.chen, shizhu.he, kliu, jzhao}@nlpr.ia.ac.cn

Abstract

Frames defined in FrameNet (FN) share highly similar structures with events in ACE event extraction program. An event in ACE is composed of an event trigger and a set of arguments. Analogously, a frame in FN is composed of a lexical unit and a set of frame elements, which play similar roles as triggers and arguments of ACE events respectively. Besides having similar structures, many frames in FN actually express certain types of events. The above observations motivate us to explore whether there exists a good mapping from frames to event-types and if it is possible to improve event detection by using FN. In this paper, we propose a global inference approach to detect events in FN. Further, based on the detected results, we analyze possible mappings from frames to event-types. Finally, we improve the performance of event detection and achieve a new state-of-the-art result by using the events automatically detected from FN.

1 Introduction

In the ACE (Automatic Context Extraction) event extraction program, an event is represented as a structure consisting of an event trigger and a set of arguments. This paper tackles with the event detection (ED) task, which is a crucial component in the overall task of event extraction. The goal of ED is to identify event triggers and their corresponding event types from the given documents.

FrameNet (FN) (Baker et al., 1998; Fillmore et al., 2003) is a linguistic resource storing considerable information about lexical and predicate-argument semantics. In FN, a frame is defined as a composition of a Lexical Unit (LU) and a set of Frame Elements (FE). Most frames contain a set of exemplars with annotated LUs and FEs (see Figure 2 and Section 2.2 for details).

From the above definitions of events and frames, it is not hard to find that the frames defined in FN share highly similar structures as the events defined in ACE. Firstly, the LU of a Frame plays a similar role as the trigger of an event. ACE defines the trigger of an event as the word or phrase which most clearly expresses an event occurrence. For example, the following sentence "*He **died** in the hospital.*" expresses a `Die` event, whose trigger is the word *died*. Analogously, the LU of a frame is also the word or phrase which is capable of indicating the occurrence of the expressed semantic frame. For example, the sentence "*Aeroplanes **bombed** London.*" expresses an `Attack`[1] frame, whose LU is the word *bombed*. Secondly, the FEs of a frame also play similar roles as arguments of an event. Both of them indicate the participants involved in the corresponding frame or event. For example, in the first sentence, *He* and *hospital* are the arguments, and in the second sentence, *Aeroplanes* and *London* are the FEs.

Besides having similar structure as events, many frames in FN actually express certain types of events defined in ACE. Table 1 shows some examples of frames which also express events.

Frame	Event	Sample in FN
Attack	Attack	*Aeroplanes **bombed** London.*
Invading	Attack	*Hitler **invaded** Austria .*
Fining	Fine	*The court **fined** her $40.*
Execution	Execute	*He was **executed** yesterday.*

Table 1: Examples of frames expressing events.

The aforementioned observations motivate us to

[1]The notation of frames distinguishes from that of events by the italic decoration.

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 2134–2143,
Berlin, Germany, August 7-12, 2016.

explore: (1) whether there exists a good mapping from frames to event-types, and (2) whether it is possible to improve ED by using FN.

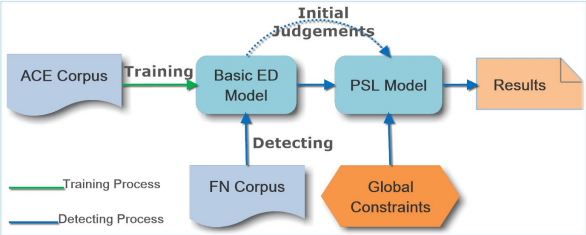

Figure 1: Our framework for detecting events in FN (including training and detecting processes).

For the first issue, we investigate whether a frame could be mapped to an event-type based on events expressed by exemplars annotated for that frame. Therefore the key is to detect events from the given exemplar sentences in FN. To achieve this goal, we propose a global inference approach (see figure 1). We firstly learn a basic ED model based on the ACE labeled corpus and employ it to yield initial judgements for each sentence in FN. Then, we apply a set of soft constraints for global inference based on the following hypotheses: 1). Sentences belonging to the same LU tend to express events of the same type; 2). Sentences belonging to related frames tend to express events of the same type; 3). Sentences belonging to the same frame tend to express events of the same type. All of the above constraints and initial judgments are formalized as first-order logic formulas and modeled by Probabilistic Soft Logic (PSL) (Kimmig et al., 2012; Bach et al., 2013). Finally, we obtain the final results via PSL-based global inference. We conduct both manual and automatic evaluations for the detected results.

For the second issue, ED generally suffers from data sparseness due to lack of labeled samples. Some types, such as `Nominate` and `Extradite`, contain even less than 10 labeled samples. Apparently, from such a small scale of training data is difficult to yield a satisfying performance. We notice that ACE corpus only contains about 6,000 labeled instances, while FN contains more than 150,000 exemplars. Thus, a straightforward solution to alleviate the data sparseness problem is to expand the ACE training data by using events detected from FN. The experimental results show that events from FN significantly improve the performance of the event detection task.

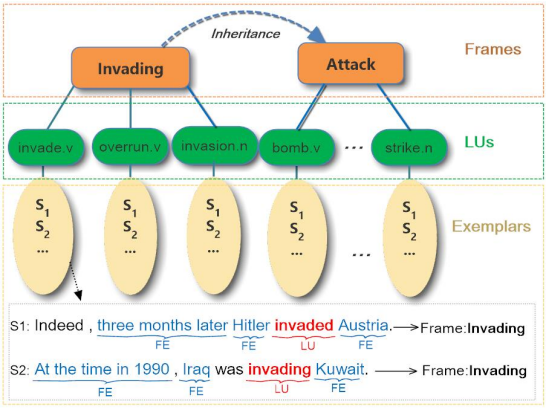

Figure 2: The hierarchy of FN corpus, where each S_k under a LU is a exemplar annotated for that LU. *Inheritance* is a semantic relation between the frames `Invading` and `Attack`.

To sum up, our main contributions are: (1) To our knowledge, this is the first work performing event detection over ACE and FN to explore the relationships between frames and events. (2) We propose a global inference approach to detect events in FN, which is demonstrated very effective by our experiments. Moreover, based on the detected results, we analyze possible mappings from frames to event-types (all the detecting and mapping results are released for further use by the NLP community[2]). (3) We improve the performance of event detection significantly and achieve a new state-of-the-art result by using events automatically detected from FN as extra training data.

2 Background

2.1 ACE Event Extraction

In ACE evaluations, an event is defined as a specific occurrence involving several participants. ACE event evaluation includes 8 types of events, with 33 subtypes. Following previous work, we treat them simply as 33 separate event types and ignore the hierarchical structure among them. In this paper, we use the ACE 2005 corpus[3] in our experiments. It contains 599 documents, which include about 6,000 labeled events.

2.2 FrameNet

The FrameNet is a taxonomy of manually identified semantic frames for English[4]. Figure 2 shows

[2]Available at https://github.com/subacl/acl16

[3]https://catalog.ldc.upenn.edu/LDC2006T06

[4]We use the latest released version, FrameNet 1.5 in this work (http://framenet.icsi.berkeley.edu).

the hierarchy of FN corpus. Listed in the FN with each frame are a set of lemmas with part of speech (i.e "invade.v") that can evoke the frame, which are called lexical units (LUs). Accompanying most LUs in the FN is a set of exemplars annotated for them. Moreover, there are a set of labeled relations between frames, such as *Inheritance*.

FN contains more than 1,000 various frames and 10,000 LUs with 150,000 annotated exemplars. Eight relations are defined between frames in FN, but in this paper we only use the following three of them because the others do not satisfy our hypotheses (see section 4.2):

Inheritance: A inherited from *B* indicates that *A* must correspond to an equally or more specific fact about *B*. It is a directional relation.

See_also: A and *B* connected by this relation indicates that they are similar frames.

Perspective_on: A and *B* connected by this relation means that they are different points-of-view about the same fact (i.e. `Receiving` vs. `Transfer`).

2.3 Related Work

Event extraction is an increasingly hot and challenging research topic in NLP. Many approaches have been proposed for this task. Nearly all the existing methods on ACE event task use supervised paradigm. We further divide them into feature-based methods and representation-based methods.

In feature-based methods, a diverse set of strategies has been exploited to convert classification clues into feature vectors. Ahn (2006) uses the lexical features(e.g., full word), syntactic features (e.g., dependency features) and external-knowledge features(WordNet (Miller, 1995)) to extract the event. Inspired by the hypothesis of One Sense Per Discourse (Yarowsky, 1995), Ji and Grishman (2008) combined global evidence from related documents with local decisions for the event extraction. To capture more clues from the texts, Gupta and Ji (2009), Liao and Grishman (2010) and Hong et al. (2011) proposed the cross-event and cross-entity inference for the ACE event task. Li et al. (2013) proposed a joint model to capture the combinational features of triggers and arguments. Liu et al. (2016) proposed a global inference approach to employ both latent local and global information for event detection.

In representation-based methods, candidate event mentions are represented by embedding, which typically are fed into neural networks. Two similarly related work has been proposed on event detection (Chen et al., 2015; Nguyen and Grishman, 2015). Nguyen and Grishman (2015) employed Convolutional Neural Networks (CNNs) to automatically extract sentence-level features for event detection. Chen et al. (2015) proposed dynamic multi-pooling operation on CNNs to capture better sentence-level features.

FrameNet is a typical resource for frame-semantic parsing, which consists of the resolution of predicate sense into a frame, and the analysis of the frame's participants (Thompson et al., 2003; Giuglea and Moschitti, 2006; Hermann et al., 2014; Das et al., 2014). Other tasks which have been studied based on FN include question answering (Narayanan and Harabagiu, 2004; Shen and Lapata, 2007), textual entailment (Burchardt et al., 2009) and paraphrase recognition (Padó and Lapata, 2005). This is the first work to explore the application of FN to event detection.

3 Basic Event Detection Model

Alike to existing work, we model event detection (ED) as a word classification task. In the ED task, each word in the given sentence is treated as a candidate trigger and the goal is to classify each of these candidates into one of 34 classes (33 event types plus a NA class). However, in this work, as we assumed that the LU of a frame is analogical to the trigger of an event, we only treat the LU annotated in the given sentence as a trigger candidate. Each sentence in FN only contains one candidate trigger, thus "the candidate" denotes both the candidate trigger of a sentence and the sentence itself for FN in the remainder of this paper. Another notable difference is that we train the detection model on one corpus (ACE) but apply it on another (FN). That means our task is also a **cross-domain** problem. To tackle with it, our basic ED approach follows representation-based paradigm, which has been demonstrated effective in the cross-domain situation (Nguyen and Grishman, 2015).

3.1 Model

We employ a simple three-layer (a input layer, a hidden layer and a soft-max output layer) Artificial Neural Networks (ANNs) (Hagan et al., 1996) to model the ED task. In our model, adjacent layers are fully connected.

Word embeddings learned from large amount of unlabeled data have been shown to be able to capture the meaningful semantic regularities of words

(Bengio et al., 2003; Erhan et al., 2010). This paper uses unsupervised learned word embeddings as the source of base features. We use the Skip-gram model (Mikolov et al., 2013) to learn word embeddings on the NYT corpus[5].

Given a sentence, we concatenate the embedding vector of the candidate trigger and the average embedding vector of the words in the sentence as the input to our model. We train the model using a simple optimization technique called stochastic gradient descent (SGD) over shuffled mini-batches with the Adadelta update rule (Zeiler, 2012). Regularization is implemented by a dropout (Kim, 2014; Hinton et al., 2012). The experiments show that this simple model is surprisingly effective for event detection.

4 Event Detection in FrameNet

To detect events in FN, we first learned the basic ED model based on ACE labeled corpus and then employ it to generate initial judgements (possible event types with confidence values) for each sentence in FN. Then, we apply a set of constraints for global inference based on the PSL model.

4.1 Probabilistic Soft Logic

PSL is a framework for collective, probabilistic reasoning in relational domains (Kimmig et al., 2012; Bach et al., 2013). Similar to Markov Logic Networks (MLNs) (Richardson and Domingos, 2006), it uses weighted first-order logic formulas to compactly encode complex undirected probabilistic graphical models. However, PSL brings two remarkable advantages compared with MLNs. First, PSL relaxes the boolean truth values of MLNs to continuous, soft truth values. This allows for easy integration of continuous values, such as similarity scores. Second, PSL restricts the syntax of first order formulas to that of rules with conjunctive bodies. Together with the soft truth values constraint, the inference in PSL is a convex optimization problem in continuous space and thus can be solved using efficient inference approaches. For further details, see the references (Kimmig et al., 2012; Bach et al., 2013).

4.2 Global Constraints

Our global inference approach is based on the following three hypotheses.

[5]https://catalog.ldc.upenn.edu/LDC2008T19

H1: Same Frame Same Event
This hypothesis indicates that sentences under the same frame tend to express events of the same type. For example, all exemplars annotated for the frame `Rape` express events of type `Attack`, and all sentences under the frame `Clothing` express NA (none) events. With this hypothesis, sentences annotated for the same frame help each other to infer their event types during global inference.

H2: Related Frame Same Event
This hypothesis is an extension of *H1*, which relaxes "the same frame" constraint to "related frames". In this paper, frames are considered to be related if and only if they are connected by one of the following three relations: *Inheritance*, *See_also* and *Perspective_on* (see section 2.2). For example, the frame `Invading` is inherited from `Attack`, and they actually express the same type of event, `Attack`. With this hypothesis, sentences under related frames help each other to infer their event types during global inference.

The previous two hypotheses are basically true for most frames but not perfect. For example, for the frame `Dead_or_alive`, only a few of the sentences under it express `Die` events while the remainder do not. To amend the this flaw, we introduce the third hypothesis.

H3: Same LU Same Event
This hypothesis indicates that sentences under the same LU tend to express events of the same type (as a remind, LUs are under frames). It is looser than the previous two hypotheses thus holds true in more situations. For example, *H3* holds true for the frame `Dead_or_alive` which violates *H1* and *H2*. In FN, LUs annotated for that frame are *alive.a, dead.a, deceased.a, lifeless.a, living.n, undead.a* and *undead.n*. All exemplars under *dead.a, deceased.a* and *lifeless.a* express `Die` events. Therefore, this hypothesis amends the flaws of the former two hypotheses.

On the other hand, the first two hypotheses also help *H3* in some cases. For example, most of the sentences belonging to the LU *suit.n* under the frame `Clothing` are misidentified as `Sue` events due to the ambiguity of the word "suit". However, in this situation, *H1* can help to rectify it because the majority of LUs under `Clothing` are not ambiguous words. Thus, under the first hypothesis, the misidentified results are expected to be corrected by the the results of other exemplars belonging to `Clothing`.

4.3 Inference

To model the above hypotheses as logic formulas in PSL, we introduce a set of predicates (see Table 2), which are grouped into two categories: observed predicates and target predicates. Observed predicates are used to encode evidences, which are always assumed to be known during the inference, while target predicates are unknown and thus need to be predicted.

$CandEvt(c,t)$ is introduced to represent $conf(c,t)$, which is the confidence value generated by the basic ED model for classifying the candidate c as an event of the type t. $SameFr(c_1,c_2)$ indicates whether the candidates c_1 and c_2 belong to the same frame. It is initialized by the indicator function $I_{sf}(c_1,c_2)$, which is defined as follows:

$$I_{sf}(c_1,c_2) = \begin{cases} 1 & c_1, c_2 \text{ from the same frame} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$SameLU(c_1,c_2)$ is similar, but applies for candidates under the same LU. The last three observed predicates in Table 2 are used to encode the aforementioned semantic relations between frames. For example, $Inherit(c_1,c_2)$ indicates whether the frame of c_1 is inherited from that of c_2, and it is initialized by the indicator function $I_{ih}(c_1,c_2)$, which is set to 1 if and only if the frame of c_1 is inherited from that of c_2, otherwise 0. $Evt(c,t)$ is the only target predicate, which indicates that the candidate c triggers an event of type t.

Type	Predicate	Assignment
Observed	$CandEvt(c,t)$	$conf(c,t)$
	$SameFr(c_1,c_2)$	$I_{sf}(c_1,c_2)$
	$SameLU(c_1,c_2)$	$I_{sl}(c_1,c_2)$
	$Inherit(c_1,c_2)$	$I_{ih}(c_1,c_2)$
	$SeeAlso(c_1,c_2)$	$I_{sa}(c_1,c_2)$
	$Perspect(c_1,c_2)$	$I_{pe}(c_1,c_2)$
Target	$Evt(c,t)$	—

Table 2: Predicates and their initial assignments.

Putting all the predicates together, we design a set of formulas to apply the aforementioned hypotheses in PSL (see Table 3). Formula f_1 connects the target predicate with the initial judgements from the basic ED model. Formulas f_2 and f_3 respectively encode *H1* and *H3*. Finally, the remaining formulas are designed for various relations between frames in *H2*. We tune the formulas's weights via grid search (see Section 5.4). The inference results provide us with the most likely

	Formulas
f_1	$CandEvt(c,t) \rightarrow Evt(c,t)$
f_2	$SameFr(c_1,c_2) \wedge Evt(c_1,t) \rightarrow Evt(c_2,t)$
f_3	$SameLU(c_1,c_2) \wedge Evt(c_1,t) \rightarrow Evt(c_2,t)$
f_4	$Inherit(c_1,c_2) \wedge Evt(c_1,t) \rightarrow Evt(c_2,t)$
f_5	$SeeAlso(c_1,c_2) \wedge Evt(c_1,t) \rightarrow Evt(c_2,t)$
f_6	$Perspect(c_1,c_2) \wedge Evt(c_1,t) \rightarrow Evt(c_2,t)$

Table 3: Formulas in the PSL model

interpretation, that is, the soft-truth values of the predicate Evt. The final detected event type t of candidate c is decided by the the equation:

$$t = \underset{t'}{\operatorname{argmax}}\, Evt(c,t') \quad (2)$$

5 Evaluations

In this section, we present the experiments and the results achieved. We first manually evaluate our novel PSL-based ED model on the FN corpus. Then, we also conduct automatic evaluations for the events detected from FN based on ACE corpus. Finally, we analyze possible mappings from frames/LUs to event types.

5.1 Data

We learned the basic ED model on ACE2005 dataset. In order to evaluate the learned model, we followed the evaluation of (Li et al., 2013): randomly selected 30 articles from different genres as the development set, and we subsequently conducted a test on a separate set of 40 ACE 2005 newswire documents. We used the remaining 529 articles as the training data set.

We apply our proposed PSL-based approach to detect events in FrameNet. Via collecting all exemplars annotated in FN, we totally obtain 154,484 sentences for detection.

5.2 Setup and Performance of Basic Model

We have presented the basic ED model in Section 3. Hyperparameters were tuned by grid search on the development data set. In our experiments, we set the size of the hidden layer to 300, the size of word embedding to 200, the batch size to 100 and the dropout rate to 0.5.

Table 4 shows the experimental results, from which we can see that the three-layer *ANN* model is surprisingly effective for event detection, which even yields competitive results compared with *Nguyen's CNN* and *Chen's DMCNN*. We believe the reason is that, compared with *CNN* and *DMCNN*,

Methods	Pre	Rec	F1
Nguyen's CNN (2015)	71.8	**66.4**	69.0
Chen's DMCNN (2015)	75.6	63.6	69.1
Liu's Approach (2016)	75.3	64.4	**69.4**
ANN (ours)	**79.5**	60.7	68.8
ANN-Random (ours)	**81.0**	49.5	61.5

Table 4: Performance of the basic ED model. *ANN* uses pre-trained word embeddings while *ANN-Random* uses randomly initialized embeddings.

ANN focuses on capturing lexical features which have been proved much more important than sentence features for the ED task by (Chen et al., 2015). Moreover, our basic model achieves much higher precision than state-of-the-art approaches (79.5% vs. 75.6%).

We also investigate the performance of the basic ED model without pre-trained word embeddings[6] (denoted by *ANN-Random*). The result shows that randomly initialized word embeddings decrease the F_1 score by 7.3 (61.5 vs. 68.8). The main reasons are: 1). ACE corpus only contains 599 articles, which are far insufficient to train good embeddings. 2). Words only existing in the test dataset always retain random embeddings.

5.3 Baselines

For comparison, we designed four baseline systems that utilize different hypotheses to detect events in FN.

(1) *ANN* is the first baseline, which directly uses a basic ED model learned on ACE training corpus to detect events in FN. This system does not apply any hypotheses between frames and events.

(2) *SameFrame (SF)* is the second baseline system, which applies *H1* over the results from *ANN*. For each frame, we introduce a score function $\phi(f,t)$ to estimate the probability that the frame f could be mapped to the event type t as follows:

$$\phi(f,t) = \frac{1}{||S_f||} \sum_{c \in S_f} I(c,t) \quad (3)$$

where S_f is the set of sentences under the frame f; $I(c,t)$ is an indicator function which is true if and only if *ANN* predicts the candidate c as an event of type t. Then for each frame f satisfying $\phi(f,t) > \alpha$, we mapped it to event type t, where α is a hyperparameter. Finally, all sentences under mapped frames are labeled as events. Note that, unlike the PSL-based approach which applies constraints as soft rules, this system utilizes *H1* as a hard constraint.

(3) *RelatedFrame (RF)* is the third baseline system, which applies *H2* over the results from *ANN*. For each frame f, we merge it and its related frames into a super frame, $f^{'}$. Similar with *SF*, a score function $\zeta(f^{'},t)$, which shares the same expression to equation 3, is introduced. For the merged frame satisfying $\zeta(f^{'},t) > \beta$, we mapped it to the event type t. Finally, all sentences under $f^{'}$ are labeled as events.

(4) *SameLU (SL)* is the last baseline, which applies the hypothesis *H3* over the results from *ANN*. Also, a score function $\psi(l,t)$ is introduced:

$$\psi(l,t) = \frac{1}{||S_l||} \sum_{c \in S_l} I(c,t) \quad (4)$$

where S_l is the set of sentences under the LU l. For each LU satisfying $\psi(l,t) > \gamma$, we mapped it to the event type t. Finally, all sentences under l are labeled as events.

5.4 Manual Evaluations

In this section, we manually evaluate the precision of the baseline systems and our proposed PSL-based approach. For fair comparison, we set α, β and γ to 0.32, 0.29 and 0.42 respectively to ensure they yield approximately the same amount of events as the first baseline system *ANN*. We tune the weights of formulas in PSL via grid search by using ACE development dataset. In details, we firstly detect events in FN under different configurations of formulas' weights and add them to ACE training dataset, respectively. Consequently, we obtain several different expanded training datasets. Then, we separately train a set of basic ED models based on each of these training datasets and evaluate them over the development corpus. Finally, the best weights are selected according to their performances on the development dataset. The weights of $f_1 \sim f_5$ used in this work are 100, 10, 100, 5, 5 and 1, respectively.

Manual Annotations

Firstly, we randomly select 200 samples from the results of each system. Each selected sample is a sentence with a highlighted trigger and a predicted event type. Figure 3 illustrates three samples. The first line of each sample is a sentence labeled with the trigger. The next line is the predicted event

[6]We thank the anonymous reviewer for this suggestion.

type of that sentence. Annotators are asked to assign one of two labels to each sample (annotating in the third line):

Y: the word highlighted in the given sentence indeed triggers an event of the predicted type.

N: the word highlighted in the given sentence does not trigger any event of the predicted type.

We can see that, it is very easy to annotate a sample for annotators, thus the annotated results are expected to be of high quality.

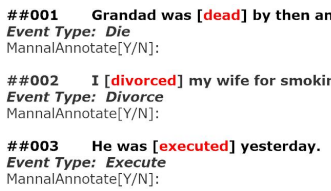

Figure 3: Examples of manual annotations.

To make the annotation more credible, each sample is independently annotated by three annotators[7] (including one of the authors and two of our colleagues who are familiar with ACE event task) and the final decision is made by voting.

Results

Table 5 shows the results of manual evaluations. Through the comparison of *ANN* and *SF*, we can see that the application of *H1* caused a loss of 5.5 point. It happens mainly because the performance of *SF* is very sensitive to the wrongly mapped frames. That is, if a frame is mismapped, then all sentences under it would be mislabeled as events. Thus, even a single mismapped frame could significantly hurt the performance. This result also proves that *H1* is inappropriate to be used as a hard constraint. As *H2* is only an extension of *H1*, *RF* performs similarly with *SF*. Moreover, *SL* obtains a gain of 2.0% improvement compared with *ANN*, which demonstrates that the "same LU" hypothesis is very useful. Finally, with all the hypotheses, the *PSL-based* approach achieves the best performance, which demonstrates that our hypotheses are useful and it is an effective way to jointly utilize them as soft constraints through PSL for event detection in FN.

5.5 Automatic Evaluations

To prepare for automatic evaluations, we respectively add the events detected from FN by each of the aforementioned five systems to ACE training corpus. Consequently, we obtain five expanded training datasets: *ACE-ANN-FN, ACE-SF-FN, ACE-RF-FN, ACE-SL-FN* and *ACE-PSL-FN*. Then, we separately train five basic ED models on each of these corpus and evaluate them on the ACE testing data set. This experiment is an indirect evaluation of the events detected from FN, which is based on the intuition that events with higher accuracy are expected to bring more improvements to the basic model.

[7]The inter-agreement rate is 86.1%

Methods		Precision (%)
Baselines	ANN	77.5
	SF	72.0
	RF	71.0
	SL	79.5
PSL-based Approach		**81.0**

Table 5: Results of manual evaluations.

Training Corpus	Pre	Rec	F_1
ACE-ANN-FN	77.2	63.5	69.7
ACE-SF-FN	73.2	64.1	68.4
ACE-RF-FN	72.6	63.9	68.0
ACE-SL-FN	77.5	64.3	70.3
ACE-PSL-FN	**77.6**	**65.2**	**70.7**

Table 6: Automatic evaluations of events from FN.

Table 6 presents the results where we measure precision, recall and F_1. Compared with *ACE-ANN-FN*, events from *SF* and *RF* hurt the performance. As analyzed in previous section, *SF* and *RF* yield quite a few false events, which dramatically hurt the accuracy. Moreover, *ACE-SL-FN* obtains a score of 70.3% in F_1 measure, which outperforms *ACE-ANN-FN*. This result illustrates the effectiveness of our "same LU" hypothesis. Finally and most importantly, consistent with the results of manual evaluations, *ACE-PSL-FN* performs the best, which further proves the effectiveness of our proposed approach for event detection in FN.

5.6 Improving Event Detection Using FN

Event detection generally suffers from data sparseness due to lack of labeled samples. In this section, we investigate the effects of alleviating the aforementioned problem by using the events detected from FN as extra training data. Our investigation is conducted by the comparison of two basic ED models, *ANN* and *ANN-FN*: the former is trained on ACE training corpus and the latter is trained on the new training corpus *ACE-PSL-FN* (introduced in the previous section), which contains 3,816 extra events detected from FN.

Methods	Pre	Rec	F_1
Nguyen's CNN(2015)	71.8	**66.4**	69.0
Chen's DMCNN(2015)	75.6	63.6	69.1
Liu's Approach(2016)	75.3	64.4	69.4
ANN (Ours)	**79.5**	60.7	68.8
ANN-FN (Ours)	77.6	65.2	**70.7**

Table 7: Effects of expanding training data using events automatically detected from FN.

Table 7 presents the experimental results. Compared with *ANN*, *ANN-FN* achieves a significant improvement of 1.9% in F_1 measure. It happens mainly because that the high accurate extra training data makes the model obtain a higher recall (from 60.7% to 65.2%) with less decrease of precision (from 79.5% to 77.6%). The result demonstrates the effectiveness of alleviating the data sparseness problem of ED by using events detected from FN. Moreover, compared with state-of-the-art methods, *ANN-FN* outperforms all of them with remarkable improvements (more than 1.3%).

5.7 Analysis of Frame-Event Mapping

In this section, we illustrate the details of mappings from frames to event types. The mapping pairs are obtained by computing the function ϕ (see Section 5.3) for each (frame, event-type) pair (f, t) based on the events detected by the PSL-based approach. Table 8 presents the top 10 mappings. We manually evaluate their quality by investigating: (1) whether the definition of each frame is compatible with its mapped event type; (2) whether exemplars annotated for each frame actually express events of its mapped event type.

For the first issue, we manually compare the definitions of each mapped pair. Except `Relational_nat_features`[8], definitions of all the mapped pairs are compatible. For the second issue, we randomly sample 20 exemplars (if possible) from each frame and manually annotate them. Except the above frame and `Invading`, exemplars of the remaining frames all express the right events. The only exemplar of `Invading` failing to express its mapped event is as follows: "*The **invasion** of China by western culture has had a number of far-reaching effects on Confucianism.*" ACE requires an `Attack` event to be a physical act, while the invasion of culture is unphysical. Thus, the above sentence does not express an `Attack` event. To sum up, the quality of our mappings is good, which demonstrates that the hypothesis *H1* is basically true.

[8]The full name is Relational_natural_relations in FN.

Frame	Event	N_e/$\|\|S_f\|\|$	ϕ
Hit_target	Attack	2/2	1.0
Relational nat_features	Meet	1/1	1.0
Invading	Attack	120/121	0.99
Fining	Fine	26/27	0.96
Being_born	Be-Born	32/36	0.88
Rape	Attack	104/125	0.83
Sentencing	Sentence	57/70	0.81
Attack	Attack	99/129	0.77
Quitting	End-Position	102/137	0.74
Notification of_charges	Charge-Indict	73/103	0.71

Table 8: Top 10 mappings from frames to event types. N_e is the number of exemplars detected as events; $||S_f||$ and ϕ hold the same meanings as mentioned in Section 5.3.

5.8 Analysis of LU-Event Mapping

This section illustrates the details of mappings from LUs to event types. The mapping pairs are obtained by computing the function ψ (see Section 5.3). Table 9 presents the top 10 mappings. In FN, each LU belongs to a frame. In table 9, we omit the frame of each LU because of space limitation[9].

LU	Event	N_e/$\|\|S_l\|\|$	ψ
gunfight.n	Attack	14/14	1.0
injure.v	Injure	14/14	1.0
divorce.n	Divorce	11/11	1.0
decapitation.n	Die	5/5	1.0
trial.n	Trial-Hearing	25/25	1.0
assault.v	Attack	21/21	1.0
fight.v	Attack	12/12	1.0
arrest.n	Arrest-Jail	38/38	1.0
divorce.v	Divorce	35/35	1.0
shoot.v	Attack	2/2	1.0

Table 9: Top 10 mappings from LUs to event types. N_e is the number of exemplars detected as events; $||S_l||$ and ψ hold the same meanings as mentioned in Section 5.3.

To investigate the mapping quality, we manu-

[9]Their frames separately are Hostile_encounter, Cause_harm, Forming_relationships, Killing, Trial, Attack, Quarreling, Arrest, Forming_relationships and Hit_target.

ally annotate the exemplars under these LUs. The result shows that all exemplars are rightly mapped. These mappings are quite good. We believe the reason is that an LU is hardly ambiguous due to its high specificity, which is not only specified by a lemma but also by a frame and a part of speech tag. Table 9 only presents the top 10 mappings. In fact, we obtain 54 mappings in total with $\psi = 1.0$. We released all the detected events and mapping results for further use by the NLP community.

6 Conclusions and Future Work

Motivated by the high similarity between frames and events, we conduct this work to study their relations. The key of this research is to detect events in FN. To solve this problem, we proposed a PSL-based global inference approach based on three hypotheses between frames and events. For evaluation, we first conduct manual evaluations on events detected from FN. The results reveal that our hypotheses are very useful and it is an effective way to jointly utilize them as soft rules through PSL. In addition, we also perform automatic evaluations. The results further demonstrate the effectiveness of our proposed approach for detecting events in FN. Furthermore, based on the detected results, we analyze the mappings from frames/LUs to event types. Finally, we alleviate the data sparseness problem of ED by using events detected from FN as extra training data. Consequently, we obtain a remarkable improvement and achieve a new state-of-the-art result for the ED task.

Event detection is only a component of the overall task of event extraction, which also includes event role detection. In the future, we will extend this work to the complete event extraction task. Furthermore, event schemas in ACE are quite coarse. For example, all kinds of violent acts, such as street fights and wars, are treated as a single event type `Attack`. We plan to refine the event schemas by the finer-grained frames defined in FN (i.e. `Attack` may be divided into *`Terrorism`*, *`Invading`*, etc.).

Acknowledgements

This work was supported by the Natural Science Foundation of China (No. 61533018), the National Basic Research Program of China (No. 2014CB340503) and the National Natural Science Foundation of China (No. 61272332). And this work was also supported by Google through focused research awards program.

References

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8.

Stephen Bach, Bert Huang, Ben London, and Lise Getoor. 2013. Hinge-loss markov random fields: convex inference for structured prediction. In *Proceedings of 29th Annual Meeting of the Association for Uncertainty in Artificial Inteligence*, pages 1–10.

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *Proceedings of 17th Annual Meeting of the Association for Computational Linguistics*, pages 86–90.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The Journal of Machine Learning Research*, 3:1137–1155.

Aljoscha Burchardt, Marco Pennacchiotti, Stefan Thater, and Manfred Pinkal. 2009. Assessing the impact of frame semantics on textual entailment. *Natural Language Engineering*, 15(04):527–550.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of 53rd Annual Meeting of the Association for Computational Linguistics*, pages 167–176.

Dipanjan Das, Desai Chen, André FT Martins, Nathan Schneider, and Noah A Smith. 2014. Frame-semantic parsing. *Computational Linguistics*, 40(1):9–56.

Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. 2010. Why does unsupervised pre-training help deep learning? *The Journal of Machine Learning Research*, 11:625–660.

Charles J Fillmore, Christopher R Johnson, and Miriam RL Petruck. 2003. Background to framenet. *International Journal of Lexicography*, 16(3):235–250.

Ana-Maria Giuglea and Alessandro Moschitti. 2006. Shallow semantic parsing based on framenet, verbnet and propbank. *European Conference on Artificial Intelligence*, 141:563–567.

Prashant Gupta and Heng Ji. 2009. Predicting unknown time arguments based on cross-event propagation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 369–372.

Martin T Hagan, Howard B Demuth, Mark H Beale, et al. 1996. *Neural network design*. Pws Pub. Boston.

Karl Moritz Hermann, Dipanjan Das, Jason Weston, and Kuzman Ganchev. 2014. Semantic frame identification with distributed word representations. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1448–1458.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

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 49th Annual Meeting of the Association for Computational Linguistics*, pages 1127–1136.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *Proceedings of 46th Annual Meeting of the Association for Computational Linguistics*, pages 254–262.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

Angelika Kimmig, Stephen Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. 2012. A short introduction to probabilistic soft logic. In *Proceedings of the NIPS Workshop on Probabilistic Programming: Foundations and Applications*, pages 1–4.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of 51st Annual Meeting of the Association for Computational Linguistics*, pages 73–82.

Shasha Liao and Ralph Grishman. 2010. Using document level cross-event inference to improve event extraction. In *Proceedings of 48th Annual Meeting of the Association for Computational Linguistics*, pages 789–797.

Shulin Liu, Kang Liu, Shizhu He, and Jun Zhao. 2016. A probabilistic soft logic based approach to exploiting latent and global information in event classification. In *Proceedings of the thirtieth AAAI Conference on Artificail Intelligence*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

George A. Miller. 1995. Wordnet: a lexical database for english. *Communications of the Acm*, 38(11):39–41.

Srini Narayanan and Sanda Harabagiu. 2004. Question answering based on semantic structures. *International Conference on Computational Linguistics*.

Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In *Proceedings of 53rd Annual Meeting of the Association for Computational Linguistics*, pages 365–371.

Sebastian Padó and Mirella Lapata. 2005. Cross-linguistic projection of role-semantic information. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 859–866. Association for Computational Linguistics.

Matthew Richardson and Pedro Domingos. 2006. Markov logic networks. *Machine learning*, pages 107–136.

Dan Shen and Mirella Lapata. 2007. Using semantic roles to improve question answering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 12–21.

Cynthia A Thompson, Roger Levy, and Christopher D Manning. 2003. A generative model for semantic role labeling. *European Conference on Machine Learning*, pages 397–408.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of 14th Annual Meeting of the Association for Computational Linguistics*, pages 189–196.

Matthew D Zeiler. 2012. Adadelta: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.