UTD's Event Nugget Detection and Coreference System at KBP 2017

Jing Lu and **Vincent Ng**
Human Language Technology Research Institute
University of Texas at Dallas
Richardson, TX 75083-0688, USA
{ljwinnie,vince}@hlt.utdallas.edu

Abstract

We describe UTD's participating system in the event nugget detection and coreference task at TAC-KBP 2017. We designed and implemented a pipeline system that consists of three components: event nugget identification and subtyping, REALIS value identification, and event coreference resolution. We proposed using an ensemble of 1-nearest-neighbor classifiers for event nugget identification and subtyping, a 1-nearest-neighbor classifier for REALIS value identification, and a learning-based multi-pass sieve approach consisting of 1-nearest-neighbor classifiers for event coreference resolution. Though conceptually simple, our system compares favorably with other participating systems, achieving F1 scores of 50.37, 40.91, and 33.87 on these three tasks respectively on the English dataset, and F1 scores of 46.76, 35.19, and 28.01 on the Chinese dataset. In particular, it ranked first on Chinese event nugget coreference.

1 Introduction

This year UTD participated in the event nugget detection and coreference task at TAC-KBP 2017. The task aims to identify (1) the explicit mentioning of events in text for three languages (English, Chinese and Spanish); (2) the event types/subtypes and three REALIS values for each event mention following the Rich ERE annotation standard; and (3) all full event coreference links. We participated in this task for English and Chinese.

In this paper, we present the system we developed for this task. We designed and implemented a pipeline system that consists of three components: event nugget identification and subtyping, REALIS value identification and event coreference. We describe each of them in detail in Section 2. The results of official evaluation are shown in Section 3.

2 UTD's System

In this section, we describe our system, which operates in three steps. First, it performs event nugget identification and subtyping, which involves detecting all explicit mentioning of events with certain specified types in text (Section 2.1). Second, it performs REALIS value identification on the event mentions extracted in the first step (Section 2.2). Third, it performs event coreference resolution on the event mentions extracted in the first step (Section 2.3).

2.1 Event nugget Identification and Subtyping

We employ multiple 1-nearest neighbor models for event nugget identification and subtyping. In each model, different features are used to represent an instance. To identify event mentions and their subtypes in a document, we first apply the 1-nearest neighbor models independently to the document. Then, we collect the union of event mentions and their subtypes identified by each model. If an event mention is classified as subtype A by model i and subtype B by model j, we collect both subtypes in the final result. In this way, we can assign multiple subtypes to each event mention.

To train the English system, we use each single word as a training instance. Additionally, we use as training instances those phrases that are true triggers according to the training data. If the word or phrase is not a trigger, the class label of the corresponding training instance is None. We create test instances from (1) the words and phrases in the test documents that also appeared in the training data as true triggers, as well as (2) all the verbs and nouns in the test documents. We apply each model to a test instance as follows. First, we pick the training instances whose lemmatized triggers are the same as the lemmatized trigger of the test instance as its neighbors. Then, we use Jaccard to measure the distance between the test instance and each of its neighbors identified in the previous step.

We implement five 1-nearest neighbor models for English system: **Model 1:** For candidate triggers that are verbs, we use the entity type of their subjects and objects as features, where the subjects and objects are extracted from the dependency parse trees obtained using the Stanford CoreNLP toolkit (Manning et al., 2014). For candidate triggers that are nouns, we employ heuristics to extract their agents and patients and use their entity type as features. These entity types are obtained from Stanford CoreNLP NER tagger. **Model 2:** For candidate triggers that are verbs, we use the head words of their subjects and objects as features. For candidate triggers that are nouns, we use the head words of their heuristically extracted agents and patients as features. **Model 3:** We use the WordNet synset ids of the candidate trigger and its hypernym as features. **Model 4:** We use the entity types of the syntactically/physically nearest entity to the trigger in syntax parse tree as features. **Model 5:** We use the unigrams in the sentence in which the candidate trigger appears as features.

The Chinese system is similar to its English counterpart. We follow the strategy used in Chen and Ng's (2012) Chinese event extraction system to generate training and test instances. Specifically, we use each single word as a training instance and assign its class label as its gold subtype or None. To create test instances, we posit a word in a test document as a test instance if it appears in a training document as a true event trigger or if it contains a character that appears within a verb trigger in the training set. We implement five 1-nearest neighbor models for the Chinese system: **Models 1,4 and 5**: they are the same as those used in the English system. **Model 2:** We use the characters of the candidate trigger and the entry number of the candidate trigger in a Chinese synonym dictionary as features.[1] **Model 3:** For candidate triggers that are verbs, we use the head words of their subjects and objects as features. For candidate triggers that are nouns, we use the head words of their heuristically extracted agents and patients as features.

2.2 REALIS value identification

This component determines the REALIS value for each event mention, each of which is created from a candidate trigger extracted in the previous step. We implement a 1-nearest neighbor model. For each test instance, we pick the training instances who have the same lemmatized triggers and subtype as its neighbors. Then, we use Jaccard to measure the distance between the test instance and each of its neighbors. We use the following features to represent each training and testing instance: the part-of-speech (POS) of the trigger; the auxiliary verb of a verb trigger and its POS tag; the negative word and its POS tag if it exists in the clause; the main verb within the clause containing the trigger word and its POS tag.

2.3 Event Coreference Resolution

We employ a multi-pass sieve approach to event coreference resolution. Each sieve is composed of a 1-nearest neighbor model for classifying whether two event mentions are coreferent or not. Sieves are ordered by their precision, with the most precise sieve appearing first. To resolve a set of event mentions in a document, the resolver makes multiple passes over them: in the i-th pass, it uses only the 1-nearest neighbor model in the i-th sieve to find an antecedent for each event mention. The candidate antecedents are ordered by their positions in the document. The partial clustering of event mentions generated in the i-th sieve is then passed to the i+1-th sieve. Specifically, the i+1-th sieve will not classify event mention pairs which are already classified as coreferent in the earlier sieves. In this way, later passes can exploit the information computed by previous passes, but the decisions made earlier cannot be overridden later.

[1] The Chinese synonym dictionary is HIT-SCIR's Tongyici cilin (extended).

We use the pairs of event mentions that have the same subtype as training instances. For each test document, we generate pairs of event mentions that have the same subtype, where subtype information was determined by the trigger component described in Section 2.1. In each sieve, the unigrams of the two sentences containing the two triggers involved are used as features. We use Jaccard to measure the distance between a pair of instances.

In each sieve, we use different strategies to choose the neighbors of each test instance. The English resolver and the Chinese resolver both employ the same two sieves described below:

Sieve 1: Given a test mention pair, we choose as its neighbors those training mention pairs that satisfy the following conditions: (1) their lemmatized triggers are the same as the lemmatized trigger pair of test mention pair; (2) their trigger subtype is the same as that of the test mention pair; and (3) the sentence distance d_{train} between the two mentions in a training mention pair must be in the range [d_{test}-m_1, d_{test}+m_1], where d_{test} is the sentence distance between the two mentions in the test mention pair, and m_1 is a tunable parameter.

Sieve 2: This sieve only classifies a test mention pair if the two triggers it contains have the same lemma. Given a test mention pair, we choose as its neighbors those training mention pairs where their triggers have the same lemma, their trigger subtype is the same as that of the test mention pair, and the sentence distance d_{train} is in the range [d_{test}-m_2, d_{test}+m_2].

Also in each sieve, we implement heuristics to find out test mention pairs with incompatible subjects or objects and then force those test mention pairs to be not coreferent. If the subject pair or object pair of a test mention pair satisfy any of the following conditions, we consider them as incompatible: (1) we calculate the entity coreference probability from the training data. If the probability of the subject pair or object pair to be coreferent are below a certain threshold, they are not compatible. The threshold is tuned on the development set. (2) If the subject pair or object pair are named entities, but their NER labels are not matched, they are not compatible.

3 Evaluation

3.1 Data

For the English system, we use LDC2015E29, LDC2015E68, LDC2015E73, LDC2015E94 and LDC2016E72 as training datasets. For the Chinese system, we use LDC2015E78, LDC2015E105, LDC2015E112 and LDC2016E72 as training datasets. For both systems, 80% of the documents are used for model training, and the remaining 20% are used for development, specifically for tuning parameters m_i in the event coreference resolution component and the threshold for the heuristic. All three components are evaluated on LDC2017E51. We only evaluate on the 18 event subtypes selected by the KBP 2017 organizers.

3.2 Evaluation Metrics

We report event nugget detection performance in terms of recall, precision and F-score for four nugget detection metrics, namely span, mention subtype only, REALIS value only and joint metric for span, mention subtype and REALIS value.

To evaluate event coreference performance, we employ four commonly-used coreference scoring measures as implemented in the official scorer provided by the KBP 2017 organizers, namely MUC (Vilain et al., 1995), B^3 (Bagga and Baldwin, 1998), CEAF_e (Luo, 2005) and BLANC (Recasens and Hovy, 2011). Each of these evaluation measures reports results in terms of recall, precision, and F-score.

3.3 Results and Analysis

Table 1 shows the results of event nugget detection, which includes the first two steps of our pipeline system. For nugget identification and subtyping, we achieve F-scores of 50.37 on the English dataset and 46.76 on the Chinese dataset. When examining the result of each type, we find that events of types "Contact" have lower performance, especially lower recall. In the discussion forum documents, our system failed to identify a lot of event mentions with subtype "contact.correspondence". Many event mentions with this subtype are just discussions between threads. Without understanding the context but just the sentence, it is difficult to find triggers. For example, just from the sentence "Thanks for

Metric	English			Chinese		
	Prec	Rec	F1	Prec	Rec	F1
Span	61.74	57.66	59.63	52.69	53.02	52.85
Subtype	52.16	48.71	50.37	46.61	46.91	46.76
REALIS	42.36	39.56	40.91	35.08	35.30	35.19
All	35.01	32.70	33.81	31.07	31.27	31.17

Table 1: Event Nugget Detection performance on the KBP 2017 official evaluation.

Metric	English						Chinese					
	Run 1			**Run 2**			**Run 1**			**Run 2**		
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
B^3	39.60	39.55	39.57	40.60	39.18	39.88	34.99	33.42	34.18	35.63	33.02	34.28
CEAF$_e$	35.76	34.66	35.20	35.68	35.78	35.73	30.76	33.82	32.22	30.86	34.65	32.64
MUC	36.73	31.58	33.96	38.29	30.24	33.79	29.89	24.73	27.07	30.08	23.52	26.40
BLANC	28.47	23.78	25.91	29.24	23.53	26.06	21.03	17.04	18.57	21.13	16.79	18.40
	Average = 33.66			**Average = 33.87**			**Average = 28.01**			Average = 27.93		

Table 2: Event Coreference Resolution performance on the KBP 2017 official evaluation.

quick reply, what shipping company did you use? ", little information can be used to identify "reply" as the trigger. Also, our system failed to distinguish between mentions with subtype "contact.broadcast" and "contact.contact", especially those with a trigger "said". Another source of precision error can be attributed to the mismatch in the class distributions between training and testing set. For example, in the training set, all mentions with trigger "rally" are annotated as having subtype "conflict.demonstrate". But in the test set, the mentions with trigger "rally" are annotated as "contact.broadcast".

For the REALIS value identification component, we achieve F-scores of 40.91 on the English dataset and 35.19 on the Chinese dataset. A closer examination of the results reveals that some conditional events that should have the value "Other" are misclassified as "Actual". Also, some events with simple present tense should be "Actual" but are misclassified as "Other". Additional work should be performed to disambiguate these cases.

For the event coreference resolution task, we submitted the following two runs:

Run 1: The resolver employs all two sieves without heuristics.

Run 2: The resolver employs all two sieves with heuristics.

Table 2 shows the results of our event coreference resolution system. The best English result is obtained from Run 2, where we achieve an average F-score of 33.87. The best Chinese result is obtained from Run 1, where we achieve an average F-score of 28.01.

The major source of precision error can be attributed to the fact that our system tends to posit all event mentions having the same trigger word as coreferent to form a long coreference link. The major source of recall error can be attributed to unseen coreferent trigger pairs. Because of the way we choose neighbors in the 1-nearest neighbor model, a test mention pair will not have any neighbors and will therefore not be posited as coreferent if its trigger pair is unseen in the original training data. The final source of recall error can be attributed to the missing triggers. For both languages, the trigger classifier failed to identify trigger words/phrases that are unseen or rarely-occurring in the training data. As a result of these missing triggers, many event coreference links cannot be established.

4 Conclusion

We presented UTD's participating system in the 2017 TAC-KBP event nugget detection and coreference task. We implemented a pipeline system that first identified event triggers and their subtypes using multiple 1-nearest neighbor models, then classified the REALIS value and finally employed a multi-pass sieve approach to identify event coreference links. Our system ranked first in Chinese event nugget coreference.

References

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *Proceedings of the Linguistic Coreference Workshop at The First International Conference on Language Resources and Evaluation*, pages 563–566.

Chen Chen and Vincent Ng. 2012. Joint modeling for Chinese event extraction with rich linguistic features. In Proceedings of the 24th International Conference on Computational Linguistics, pages 529–544.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In Proceedings of the Human Langauge Technology Coreference and the Conference on Empirical Methods in Natural Language Processing, pages 25–32.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 55–60.

Marta Recasens and Eduard Hovy. 2011. *BLANC: Implementing the Rand Index for Coreference Evaluation*. Natural Language Engineering, pages 485–510.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In Proceedings of the Sixth Message Understanding Conference, pages 45–52.