UTD's Event Nugget Detection and Coreference System at KBP 2016

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 2016. 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, an SVM 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 46.99, 39.78, and 30.08 on these three tasks respectively on the English dataset, and F1 scores of 40.01, 33.68, and 26.43 on the Chinese dataset. In particular, it ranked first on English event nugget detection as well as on English and Chinese event nugget coreference.

1 Introduction

This year UTD participated in the event nugget detection and coreference task at TAC-KBP 2016. 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 four 1-nearest neighbor models for English system: **Model 1:** For candidate triggers that are verbs, we use the head words 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 head words as features. **Model 2:** For candidate triggers that are verbs, we use the entity types of their subjects and objects as features. For candidate triggers that are nouns, we use the entity types of their heuristically extracted agents and patients as features. These entity types are provided by an entity typing classifier trained on corpora LDC2015E68 and LDC2015E29, both of which are annotated with Rich ERE entities. **Model 3:** We use the WordNet synset ids of the candidate trigger and its hypernym as features. **Model 4:** 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 and 2**: they are the same as those used in the English system. The entity types are provided by an entity typing classifier trained on corpora LDC2015E78, LDC2015E105 and LDC2015E112. **Model 3:** We use the head word of the entity that is syntactically/textually closest to the candidate trigger as features. **Model 4:** 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 5:** We use the entity type of the entity that is syntactically/textually closest to the candidate trigger 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 train one multi-class SVM classifier using the $SVM^{multiclass}$ software package (Tsochantaridis et al., 2004). We create one instance for each event mention. To represent each training/test instance, we use following features, which can be divided into two groups:

Group 1 (Event Mention features). The three features encode: the trigger word of the event mention; the part-of-speech (POS) of the trigger; and the event subtype of the trigger.

Group 2 (Syntactic features). The six features encode: the path from the leaf node of the trigger to its governing clause; the main verb within the clause containing the trigger word and its POS tag; a Boolean feature indicating whether a negative word exists in the clause containing the trigger word; the auxiliary verb of a verb trigger 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

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

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.

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 three 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].

Sieve 3: This sieve utilizes additional positive training mention pairs across documents that are created as follows. For example, suppose an event mention having trigger 1 and an event mention having trigger 2 are coreferent in document A. In addition, suppose that an event mention having trigger 2 and an event mention having trigger 3 are coreferent in document B. If the event mention having trigger 1 and the event mention having trigger 3 are not coreferent in any training document, we create a new positive training mention pair. Using this augmented training set, we apply the same strategy to choose neighbors as in Sieve 1. We tune a different parameter m_3 for the third condition.

3 Evaluation

3.1 Data

For the English system, we use LDC2015E29, LDC2015E68, LDC2015E73 and LDC2015E94 as training datasets. For the Chinese system, we use LDC2015E78, LDC2015E105 and LDC2015E112 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. All three components are evaluated on LDC2016E72. We only evaluate on the 18 event subtypes selected by the KBP 2016 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 2016 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 46.99 on the English dataset and 40.01 on the Chinese dataset. When examining the result of each type, we find that events of types Man-

Metric	English			Chinese		
	Prec	Rec	F1	Prec	Rec	F1
Span	55.36	53.85	54.59	47.23	43.16	45.10
Subtype	47.66	46.35	46.99	41.90	38.29	40.01
REALIS	40.34	39.23	39.78	35.27	32.23	33.68
All	34.05	33.12	33.58	31.76	29.02	30.33

Table 1: Event Nugget Detection performance on the KBP 2016 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	34.89	39.87	37.22	35.45	39.78	37.49	33.18	32.49	32.83	35.31	30.28	32.60
CEAF$_e$	35.58	32.24	33.83	35.76	32.80	34.21	32.45	29.34	30.82	29.84	30.95	30.39
MUC	24.10	28.73	26.21	24.59	28.42	26.37	25.00	23.59	24.27	23.58	12.37	16.23
BLANC	21.33	23.68	22.10	21.62	23.51	22.25	18.45	17.33	17.80	16.64	12.00	13.83
	Average = 29.84			**Average = 30.08**			**Average = 26.43**			Average = 23.26		

Table 2: Event Coreference Resolution performance on the KBP 2016 official evaluation.

ufacture, Contact and Transaction have lower performance. One source of precision error can be attributed to multi-label classification. For example, an event mention was labeled as belonging to different subtypes of ”Contact” in different passes. However, given an event of type ”Contact”, only one subtype should be correct according to the Rich ERE annotation standard. The second source of precision error has to do with the fact that our system tends to assign the same subtype to all event mentions having the same trigger. The third source of precision error is caused by the fact that our Chinese system have difficulties with identifying triggers having one single character. One major source of recall error can be attributed to the difficulty of correctly extracting features in discussion forum documents owing to their informal writing style. Another source of recall error can be attributed to the inability of our system to identify trigger words/phrases that are unseen or rarely-occurring in the training data.

For the REALIS value identification component, we achieve F-scores of 35.27 on the English dataset and 33.68 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 three sieves.

Run 2: The resolver employs only the first two sieves.

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 30.08. The best Chinese result is obtained from Run 1, where we achieve an average F-score of 26.43. The major source of precision error can be attributed to the fact that our system tends to posit event mentions having the same trigger word as coreferent. 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 or the training data augmented with the cross-document trigger pairs in Sieve 3. An additional source of recall error has to do with the fact that our system does not tend to posit event mentions having different triggers as coreferent. 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 2016 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 English event nugget detection as well as in English and 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.

Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. 2004. Support vector machine learning for interdependent and structured output spaces. In Proceedings of the 21st International Conference on Machine Learning, pages 104–112.

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.