UTD's Event Nugget Detection and Coreference System at KBP 2015

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 system participated in the event nugget detection and coreference task at the TAC-KBP 2015. We design and implement a pipeline system that consists of three components: event trigger identification and subtyping, REALIS value identification and event coreference. In particular, we proposed a multi-pass sieve approach to event coreference resolution. UTD's system achieved F1 scores of 57.45, 45.21, and 32.36 on those three components respectively.

1 Introduction

This year UTD participated in the event nugget detection and coreference task at TAC-KBP 2015. The task aims to identify (1) the explicit mentioning of events in text for English; (2) event types/subtypes and three REALS values for each mention of event following the Rich ERE annotation standard; and (3) all full event coreference links.

In this paper, we present our system for this task. We design and implement a pipeline system that consists of three components: event trigger identification and subtyping, REALIS value identification and event coreference. We describe each of them in details in Section 2. The results of official evalution is shown in Section 3.

2 UTD's System

In this section, we describe our system, which operates in three steps. First, it performs event trigger 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 Trigger Identification and Subtyping

This component extracts event triggers and determines the semantic type and subtype of each event mention. In the KBP 2015 corpus, there are 9 event types and 38 event subtypes. A event trigger can be a single word or a multi-word phrase. We recast the task of identifying event triggers as a sequence labeling task, where we train one CRF using the CRF++ package[1]. As mentioned in the introduction, since each word can trigger multiple event mentions having different types/subtypes, we train one CRF for each type. Specifically, for classifier of type t_j, we create one instance for each word w_i, assigning it a class label that indicates whether it begins a trigger with subtype s_{jk} (B-s_{jk}), is inside a trigger with subtype s_{jk} (I-s_{jk}), begins a trigger with other types (B-$t_{m \neq j}$), is inside a trigger with other types (I-$t_{m \neq j}$) or is outside a trigger (O). So there are ($2\times$ number of subtypes of t_j + $2\times$ number of other types $+1$) labels in total. Below we describe the features used to represent w_i, which can be divide into three categories: lexical, syntactic and semantic. We use Stanford CoreNLP package (Manning et al., 2014) to extract the linguistic information needed to compute these features, including the part-of-speech tags

[1] https://taku910.github.io/crfpp/

and syntactic parse trees.

Lexical: word unigrams (w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}) ; word bigrams ($w_{i-1}w_i$, w_iw_{i+1}); word trigrams ($w_{i-2}w_{i-1}w_i$, $w_{i-1}w_iw_{i+1}$, $w_iw_{i+1}w_{i+2}$), the part-of-speech tag of w_i; lemmatized word unigrams, bigrams and trigrams.

Syntactic: depth of w_i's node in its syntactic parse tree; the path from the leaf node of w_i to the root in its syntactic parse tree; the phrase structure expanded by the parent of w_i's node; the phrase type of w_i's node.

Semantic: the WordNet synset id of w_i; the WordNet synset ids of the w_i's hypernym, its parent, and its grandparent; When computing these semantic features, we only use the synset corresponding to w_i's first sense.

One exception is instances of type Contact. According to the guidelines for annotating contact events, the subtypes are decided based on the four attributes, namely formality, scheduling, medium and audience. We notice that all subtypes of contact event have same value for attribute formality and scheduling. So we train two CRFs for annotating the medium and audience attributes separately using the same features mentioned above.

We improve the recall of event mention detection in a postprocessing process as follows. First, we construct a wordlist containing triggers that appear infrequently (less than 10 times) in the training data and do not belong more than one subtype according to the training data. For example, the word "hijack" appears only a few times in the training data but is always labeled as "Conflict.Attack". Then, we extract any word as a trigger with the corresponding subtype as long as it appears in the wordlist.

2.2 REALIS value identification

This component determines the REALIS value for each event mention. We train one multi-class SVM classifier using the libSVM software package (Chang and Lin, 2001). We create one instance for each event mention. We use following features to represent each training and test instance, which can be divide into three groups:

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

Group 2 (Syntactic features). The six features encode: the main verb within the clause containing the trigger word and its POS tag; the left and right word of the main verb and their POS tags; a boolean feature indicating whether a negative word exists in the clause containing the trigger word.

Group 3 (Other features). The three features encode: the plurality of trigger if the trigger is a noun; boolean features indicating whether there are time and location entities in the clause containing the trigger.

2.3 Event Coreference Resolution

This component identifies event coreference links using a multi-pass sieve approach. The sieve approach has been successfully applied to entity coreference resolution (Raghunathan et al., 2010), but has not yet been applied to event coreference resolution.

A sieve is composed of one or more heuristic rules. Each rule extracts a coreference relation between two event mentions. 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 rules 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 pass is then passed to the i+1-th pass. In this way, later passes can exploit the information computed by previous passes, but the decision make earlier cannot be overridden later.

In our approach, later sieves exploit the decisions made by the earlier sieves as follows. When two event mentions are posited as coreferent by a sieve, any argument extracted for one mention will be shared by the other mention. It is this sharing of argument among coreferent event mentions that will be exploited by the later sieves. In our current implementation, we heuristically extract event arguments that play the roles of agents and patients. For instance, one rule posts the subject of a verb trigger as agent, and another rule posits the possessor of a noun trigger as agent.

We designed different sieves for newswire documents and discussion forum documents. The following sieves are used for event coreference resolution in newswire documents.

1. **Newswire Headline sieve:** this sieve is mo-

tivated by the journalistic nature of newswire documents. The first sentence in the newswire documents always contains a detailed explanation of the headline. This sieve posits two event mention in the headline and an event mention the first sentence as coreferent if they have the same subtype and their triggers are in the same WordNet synset.

2. **Strict Event Coreference sieve:** this sieve follows the strict event coreference criteria. Two mentions are posited as coreferent if they satisfy all of the following conditions: (a) they have the same subtypes; (b) their triggers are in the same lemmatized form; (c) their agents/patients are in the same entity coreference chain or are lexically identical (if they are non-pronominal); (d) their triggers are in the same entity coreference chain if they are nouns.

3. **Strict Trigger Match sieve:** this sieve posits two event mentions with noun triggers as coreferent if they have the same subtypes and their triggers have the same lemma and same modifiers.

4. **Semantically Similar Trigger sieve:** this sieve relaxes the Strict Event Coreference Sieve by deleting conditions (b) and (d), but it requires the triggers of the two mentions or the hypernyms of the triggers to be in the same WordNet synset.

For discussion forum documents, we employ essentially the same sieves except that we replace the first sieve with a sieve that posits two event mentions as coreferent if their triggers and the sentences containing them are identical. This sieve is motivated by the nature of a discussion forum where an author usually quotes a preceding post to which she wants to respond.

3 Evaluation

3.1 Data

We trained our system using the KBP 2015 event detection training corpus and test on the evaluation corpus. Both the event trigger subtyping system and the REALIS value identification sytem are trained on and applied to documents from both domains.

3.2 Evaluation Metrics

To evaluate event coreference performance, we employ four commonly-used coreference scoring measures as implemented in the official scorer provided by the KBP 2015 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 (R), precision (P), and F-score (F). We also report event nugget detection performance in terms of recall, precision and F-score for 4 nugget detection metrics, namely plain, mention type only, REALIS value only, and joint metric for mention type and REALIS value.

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 the trigger identification and subtyping component, we achieve an F-score of 57.45%. When examining the result of each type, we find that events of type Manufacture, Contact and Business have lower performance. One reason can be attributed to the scarcity of instances belonging to these types in the training corpus. For example, there are only 22 event mentions of type Manufacture in the training corpus. In addition, as mentioned before, instances of the subtypes of Contact are difficult to identify. Currently we use the same feature set for different attributes. We believe it would be useful to explore different features for different attributes.

	P	R	F1
plain	74.85	56.76	64.56
mention type	66.60	50.50	57.45
realis	52.41	39.75	45.21
mention type+ realis	46.00	34.88	39.67

Table 1: Event Nugget Detection performance on the KBP 2015 official evaluation

For the REALIS value identification component, we achieve an F-score of 45.21. A close examination of the results reveals that some conditional events that should have the value "Other" are misclassified as "Actual". Also, some events with the simple present tense should be "Actual" but are misclassified as "Other". Additional work should be performed on disambiguating these cases.

Table 2 shows the result of the event coreference resolution component using scorer of version 1.7 [2].

[2]Scorer can be found at http://cairo.lti.cs.cmu.edu/kbp/2015/event/scoring

	P	R	F1
MUC	46.67	13.66	21.13
B^3	62.36	33.62	43.69
$CEAF_e$	41.05	41.84	41.44
BLANC	44.26	17.18	23.18
Average			32.36

Table 2: Event Coreference Resolution performance on the KBP 2015 official evaluation

As we can see, we achieve an averaged F-score of 32.36. A major source of error stems from the system's inability to cluster events with noun triggers. The major difficulty comes from argument extraction. Different from events with verb triggers whose arguments can be extracted from common patterns among all event types, the arguments of events with noun triggers of different types usually have different indicators. For example, arguments of Conflict event mentions with noun triggers such as attack and war are commonly preceded by prepositions such as "against", whereas arguments of Transaction event mentions with noun triggers such as buyer are commonly preceded by prepositions such as "for". This problem could be addressed by training classifiers to extract role-specific arguments of event mentions belonging to different types/subtypes. Another major source of error stems from the system's tendency to cluster event mentions whose triggers have the same lemma. Although the semantically similar trigger sieve is used, more background knowledge is needed to resolve these difficult cases.

4 Conclusion

We presented UTD's system in the 2015 TAC-KBP event nugget detection and coreference task. We implemented a pipeline system that first identified event triggers and their subtypes, then classified the REALIS value and finally employed a multi-pass sieve approach to identify event coreference links.

References

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

Chang, Chih-Chung and Lin, Chih-Jen. 2011. *LIBSVM: A library for support vector machines*. ACM Transactions on Intelligent Systems and Technology,27:1–27:27.

X. Luo. 2005. *On coreference resolution performance metrics*. Proceedings of HLT/EMNLP,25–32.

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

Vilain, Marc and Burger, John and Aberdeen, John and Connolly, Dennis and Hirschman, Lynette. 1995. *A model-theoretic coreference scoring scheme*. muc6, Columbia, MD.

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

Raghunathan, Karthik and Lee, Heeyoung and Rangarajan, Sudarshan and Chambers, Nate and Surdeanu, Mihai and Jurafsky, Dan and Manning, Christopher. 2010. *A Multi-Pass Sieve for Coreference Resolution.* Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing,492–501.