Abstract

In this paper, we describe TAMU’s system submitted to the TAC KBP 2017 event nugget detection and coreference resolution task. Our system builds on the statistical and empirical observations made on training and development data. We found that modifiers of event nuggets tend to have unique syntactic distribution. Their parts-of-speech tags and dependency relations provides them essential characteristics that are useful in identifying their span and also defining their types and realis status. We further found that the joint modeling of event span detection and realis status identification performs better than the individual models for both tasks. Our simple system designed using minimal features achieved the micro-average F1 scores of 57.72, 44.27 and 42.47 for event span detection, type identification and realis status classification tasks respectively. Also, our system achieved the CoNLL F1 score of 27.20 in event coreference resolution task.

1 Introduction

The TAMU NLP group participated in the Event Nugget Track of TAC KBP 2017. The goal of this task is to identify the character span, classify type and realis status of event mentions and also link all the coreferent event mentions within the same text. We designed a pipeline of three neural network based classifiers for this task, the first detects event span and classify realis status, the second classifies event type and the third resolves event coreference links. These classifiers are based on simple lexical and syntactic features which are derived from the distinct distributional properties of event mentions.

Syntactic dependency relation of event triggers with their modifiers and governors are lately shown very effective for the task of temporal relations classification between event pairs (Choubey and Huang, 2017b; Yao et al., 2017; Cheng and Miyao, 2017) and identifying the temporal status of an event mention (Dai et al., 2017). The realis status of an event mention has a close association to its temporal status (Huang et al., 2016) and its relative position in temporal space. Motivated by the performance gain observed in recent research works on temporal relations, we analyze the distribution of modifiers of events with different realis status. Let’s look at the examples below (bold-faced words in blue are event mentions and other words in blue are their modifiers):

(1) [Actual] Continental Airlines board of directors met Wednesday to discuss a merger with United Airlines, a person familiar with the situation said.

(2) [Other] If United and Continental marry, the new airline will be the nation’s largest carrier, eclipsing Delta Airlines, which merged with Northwest Airlines in 2008.

(3) [Actual] If United and Continental marry, the new airline will be the nation’s largest carrier, eclipsing Delta Airlines, which merged with Northwest Airlines in 2008.

In examples (1) and (3), the presence of modifiers Wednesday and 2008 help in binding the events met and merged to the timeline. These temporal modifiers imply that both the events have already occurred in the past and thus should be classified as the actual event. On the other hand, the modifier if of event marry in example (2) implies that the event is hypothetical. Our analysis and empirical evaluation suggest that these dependency parse based features are also beneficial to identifying the realis status of events.
In our experiments, we further found that the event span detection performs better when modeled jointly with realis status identification. We evaluated two neural network classifiers: the first classifier is trained to predict whether the given word is an event trigger or not and the second classifier is trained to jointly predict whether the given word is an event trigger together with its realis status on the 2016 evaluation dataset. We found that the second classifier achieved around 2% higher F1 score on event span detection, with major improvement coming from the precision.

2 Motivation

![Table 1: Frequency of dependency relation with modifiers for event and non-event words](https://github.com/prafulla77/TAC-KBP-2017-Participation)

We analyzed the dependency parse of sentences and found that modifiers of event trigger word have unique syntactic distribution. They are related to the trigger word with few frequently occurring dependency relations. Moreover, they tend to have few specific parts-of-speech (POS) tags only. Based on our observations on 2015 training data, words having a modifier with a set of dependency relations like ccomp, nmod:in, nmod:tmod, nsubjpass, auxpass etc. are event triggers with very high probability. At the same time, words having modifiers attached with other relations like compound, dep, etc. are almost always non-event words (Table 1). Similar distribution also holds with the parts-of-speech tags of modifiers. While some of the POS-tags including WP, VBD, IN, TO etc. are frequently associated with event triggers, other POS-tags like EX, POS etc. are common to the non-event words (Table 2).

We further analyzed the distribution of POS-tags of words in the surface context of event and non-event words.

![Table 2: Frequency of parts-of-speech tags of modifiers for event and non-event words](https://github.com/prafulla77/TAC-KBP-2017-Participation)

We also analyzed the distribution of name entities that modify event triggers in the syntactic parse tree. Since each type of event participants can only be linked to specific event subtypes only, named entities are a strong feature for type classification. The distribution is shown in Table 4. Clearly, each type of event tends to feature certain types of entities as arguments, therefore, the presence of entities can serve as a useful evidence for event type classification.

3 System Overview

Our feature based method follows the conventional pipeline approaches which divide event nugget detection and coreference resolution into several sub-tasks. These are:

3.1 Span identification and Realis Status Classification

In the first step, we jointly perform span identification and realis status classification. We use an ensemble of neural network classifiers defined over

---

1Implementation is available at https://github.com/prafulla77/TAC-KBP-2017-Participation
Table 4: Distribution of named entities types in the context of various event subtypes (Per.- person, Loc.- location, Org.- organization, Num.- number and Misc.- miscellaneous)

| Event Type      | Per. | Loc. | Org. | Num. | Misc. |
|-----------------|------|------|------|------|-------|
| Elect           | 12   | 1    | 4    | 0    | 0     |
| Pardon          | 41   | 0    | 0    | 5    | 0     |
| Sentence        | 16   | 3    | 1    | 0    | 0     |
| Start-position  | 18   | 1    | 2    | 0    | 0     |
| End-Org.        | 0    | 0    | 3    | 0    | 0     |
| Transfer-money  | 10   | 4    | 14   | 1    | 0     |
| Transport-art.  | 0    | 13   | 1    | 0    | 0     |
| Attack          | 14   | 60   | 3    | 5    | 8     |
| Broadcast       | 39   | 11   | 28   | 1    | 0     |
| Demonstrate     | 0    | 13   | 2    | 2    | 1     |
| Transport-person| 31   | 66   | 1    | 8    | 1     |
| Contact         | 53   | 7    | 12   | 1    | 1     |
| Die             | 37   | 11   | 4    | 4    | 3     |
| Meet            | 36   | 4    | 6    | 0    | 0     |
| Acquit          | 1    | 0    | 0    | 0    | 0     |

Table 5: Features and their vector dimensions used in our span+realis and subtype classifiers (S+R- span+realis, T- type, Context features are defined over window size 2)

| Features                           | Dim. | S+R | T |
|------------------------------------|------|-----|---|
| lemma vector                       | 300  | ✓   | ✓ |
| token vector                       | 300  | ✓   | ✓ |
| POS-tag                            | 47   | ✓   | ✓ |
| context words POS-tag              | 235  | ✓   | ✓ |
| context words dependency relation   | 1040 | ✓   | ✓ |
| (token - lemma) vector             | 300  | ✓   | ✓ |
| dependency relation with modifiers  | 208  | ✓   | ✓ |
| POS-tag of modifiers               | 47   | ✓   | ✓ |
| dependency relation with governor   | 208  | ✓   | ✓ |
| POS-tag of governor                | 47   | ✓   | ✓ |
| prefix and suffix of words         | 36   | ✓   | ✓ |
| named entity type of modifiers     | 8    | ✓   | ✓ |

3.2 Event Subtype Classification

Following the strategy similar to span detection and realis classification, event subtype classifier also uses an ensemble of classifiers defined over features described in Table 5. We used KBP 2015 training and evaluation dataset to train our system. However, that dataset contains 38 event subtypes while the KBP 2017 evaluation dataset contains events from 18 subtypes only. So we model this subtask as a 19 class classification problem, where 19 classes correspond to the 18 subtypes in KBP 2017 evaluation dataset and the other. The other class means that event can be from any of the remaining 20 subtypes that are not included in evaluation dataset. Also, there are several event mentions in the dataset that have multiple subtypes. We consider only one subtype for such event mentions and ignore other subtype instances.

We trained 10 neural network classifiers for span detection and realis status identification and 3 classifiers for type classification. These classifiers differ in their architecture, training parameters and initialization. Details of all the classifiers used are described in Table 6. The configuration [2468-600-600-50-4, 0-.5-0-0-0, 10] can be interpreted as a classifier with an input layer with 2468 neurons, 3 hidden layers with 600, 600 and 50 neurons and an output layer with 4 neurons. The classifier has a dropout layer (with the dropout rate of 0.5) after first hidden layer and is trained for 10 epochs. All the classifiers use relu activation in input layer, tanh activation in all hidden layers and softmax activation in output layers.

Table 6: Span detection and realis status classifiers and type classifiers parameters. S means span+realis classifier and T means type classifier

3.3 Coreference Resolution

We replicated the pairwise within-document classifier architecture proposed in Choubey and...
Huang (2017a) for this task. The classifier uses a common neural layer shared between two event mentions that embed event lemma and parts-of-speech tags and then calculates cosine similarity, absolute and Euclidean distances between two event embeddings, corresponding to each event mention. This shared layer has 347 neurons and uses sigmoid activation function. The classifier also includes a second neural layer with 380 neurons to embed event arguments (considered named entities which modifies event mentions as the argument (Finkel et al., 2005)) that are overlapped between the two event mentions, suffix and prefix based features for both event lemmas and absolute difference between vectors of event tokens. The calculated embeddings similarities as well as the embedding of the second neural layer are concatenated and fed into the third neural layer with 10 neurons. The output activation of the third layer is finally fed into the output layer with 1 neuron that gives the confidence score to indicate the similarity between the two event mentions.\(^2\)

The second, third and output layers also use sigmoid activation function. We used 300 dimensional word embeddings (Pennington et al., 2014) and 47 dimensional one hot embeddings for pos-tags (Toutanova et al., 2003). During inference, we perform greedy merging using the classifiers predicted score. An event mention is merged to its best matching antecedent event mention if the predicted score is greater than 0.5.

4 Experiments

4.1 Dataset

The testing data of KBP 2017 consists of documents taken from the discussion forum and news articles. Therefore, we train our classifiers on both discussion forum and news articles taken from KBP 2015 training and evaluation dataset and used documents from KBP 2016 evaluation as the development dataset.

4.2 Preprocessing

We run Stanford coreNLP module for tokenization, sentence segmentation, lemmatization, POS tagging, dependency parsing, named entity recognition and coreference resolution (Manning et al., 2014; Recasens et al., 2013; Lee et al., 2011). Further, we use the cleanxml annotator available in coreNLP pipeline for removing tags and obtaining character offsets of each token. The obtained offsets are aligned to the character offset provided in the annotation files.

4.3 Performance comparison on the development dataset

In order to compare our system with the systems that participated in the event nugget detection and coreference task in KBP 2016, we evaluated our system on KBP 2016 testing dataset and used it for development and parameter tuning.

In Table 7, we illustrate the advantage of jointly modeling event span detection and realis status identification over their individual models. The results mentioned in the Table 7 are the average F1 score of 3 classifiers’ instances trained with different random initializations. From the table, it’s evident that the average performance on span detection has improved significantly when modeled together with the realis status classification. However, the performance on realis status classification remains similar.

\[
\begin{array}{llll}
\text{System} & \text{Span} & \text{Realis} \\
\text{Joint span + realis classifier} & 53.47 & 40.13 \\
\text{Separate realis and span classifiers} & 51.44 & 39.87 \\
\end{array}
\]

Table 7: Performance comparison of joint span+realis and separate span and realis classifiers on KBP 2016 evaluation dataset

In Table 8, we compare the performance of our ensemble system with its strongest and weakest member classifiers. The results show that combining multiple classifiers helped overcome the inherent problem of the neural network to over-fit according to the specific dataset. Including diverse classifiers with different dropout and network architecture helped reduce variance in the final prediction.

\[
\begin{array}{lllll}
\text{System} & \text{Span} & \text{Type} & \text{Realis} & \text{All} \\
\text{Ensemble System} & 56.14 & 44.48 & 42.59 & 33.0 \\
\text{Weakest Classifiers} & 52.44 & 41.82 & 37.16 & 29.18 \\
\text{Strongest Classifiers} & 54.03 & 43.60 & 40.28 & 31.92 \\
\end{array}
\]

Table 8: Performance comparison of our ensemble system with the best and the worst member classifiers on KBP 2016 evaluation dataset

In Table 9 and 10, we compare the performance of our complete model with the systems submitted to the KBP 2016. Our feature based classifier.

\(^2\)We implemented our classifier using the Keras library (Chollet, 2015)
compares well to the top scoring systems in KBP 2016 which modeled this task as sequence labeling problem and used complex models based on recurrent neural networks and convolutional neural networks. Specifically, compared to the best scores in KBP 2016, our model is able to achieve around 1.5% higher F1 score in event span detection task and is marginally below the best score in realis status classification task. This implies the advantage of using the dependency parse based features and joint modeling of event span detection and realis status classification subtasks.

Table 9: Performance comparison of our system on event span, type and realis status classification w.r.t. systems submitted in KBP 2016. All results are taken from Mitamura et al. (2016)

| System          | Span  | Type  | Realis | All   |
|-----------------|-------|-------|--------|-------|
| Our System      | 56.14 | 44.38 | 42.59  | 33.0  |
| Lu and Ng (2016)| 54.59 | 46.99 | 39.78  | 33.58 |
| Nguyen et al. (2016)| 54.07 | 44.38 | 42.68  | 35.24 |
| Hong et al. (2016)| 50.83 | 43.67 | 38.35  | 32.59 |
| Liu et al. (2016)| 50.49 | 44.61 | 33.11  | 29.06 |
| Zeng et al. (2016)| 49.39 | 44.47 | 36.96  | 33.1  |
| Yu et al. (2016)| 48.65 | 42.07 | 34.46  | 30.16 |
| Mihaylov and Frank (2016)| 46.85 | 32.62 | 36.83  | 26.53 |
| Wei et al. (2016)| 43.33 | 36.70 | 33.69  | 28.38 |
| Ferguson et al. (2016)| 41.25 | 34.65 | 29.75  | 25.24 |
| Satyapanich and Finin (2016)| 35.24 | 31.57 | 24.04  | 21.67 |
| Yang et al. (2016)| 39.21 | 24.77 | 21.13  | 17.87 |
| Tsai et al. (2016)| 28.07 | 21.57 | 9.70   | 7.49  |
| Dubbin et al. (2016)| 5.72  | 0.59  | 2.75   | 0.11  |

Table 10: Performance comparison of our system on coreference resolution w.r.t. systems submitted in KBP 2016. All results are taken from Mitamura et al. (2016)

| System          | Bi| CeAF| MUC | BLANC | CoNLL |
|-----------------|---|-----|-----|-------|-------|
| Our System      | 36.06 | 35.39 | 17.02 | 18.77 | 27.13 |
| Lu and Ng (2016)| 37.49 | 34.21 | 26.37 | 22.25 | 30.08 |
| Liu et al. (2016)| 35.05 | 30.45 | 24.60 | 18.79 | 27.23 |
| Nguyen et al. (2016)| 34.62 | 33.33 | 27.01 | 18.31 | 27.07 |
| Yu et al. (2016)| 20.96 | 16.14 | 17.32 | 10.67 | 16.27 |
| Yang et al. (2016)| 19.74 | 16.13 | 16.05 | 8.92  | 15.21 |
| Tsai et al. (2016)| 11.92 | 11.54 | 4.34  | 3.10  | 7.73  |

5 Evaluation on KBP 2017 dataset

We submitted 3 runs of our system for the official evaluation. They are:

Run-I: used ensemble of classifiers without any parameter tuning.
Run-II: same as Run-I with parameters tuned to produce the best result on 2016 Evaluation dataset.
Run-III: used the strongest member classifier from all the classifiers used for event span, realis status and type classification in Run-I. The coreference resolution classifier is same in all three runs.

Comparison of results of 3 runs (Tables 11 and 12) shows mixed performance. However, our hypotheses are consistent. They key observations are:

1. Run I achieves the highest F1 score for span detection, realis status classification and reals + type classification. This model doesn’t use any form of tuning on the development dataset. We can argueably conclude that inference made by aggregating multiple diverse classifiers can reduce dependency on the training parameters like dropout rates, layers etc. in the Neural Networks.

2. The events extracted in run II achieved the best coreference performance. This can be justified by the coreference evaluation setup, which requires the coreferent event mention to have the same event type. Run II has significantly higher precision for all the subtasks- span, type and realis.

3. Similar to the results on development dataset, ensemble based system (run I) performs better than the system relying on single classifier for each subtask (run III).

5.1 Macro analysis of Results

The KBP 2017 evaluation dataset contains two types of documents- discussion forum and news articles. While news articles are well structured, discussion forum articles are informal and noisy. Discussion forum articles tend to contain unnecessary punctuations, or sometimes omit punctuations and have several grammatical and spelling mistakes. Since our classifiers rely heavily on the features derived from syntactic parse, we separately analyzed the performance of our system on the discussion forum and news articles. Figures 1 and 2 shows the histogram of documents vs F1 score for span detection and type + realis status classification subtasks. It is quite interesting to find here that our system performed significantly better for the news articles compared to the noisy discussion forum articles. The lower performance of our systems on discussion forum articles can be partially accounted to the error generated from the preprocessing step. We manually analyzed the output from our preprocessing steps and observed that incorrect sentence segmentation is the dominant source of errors in most of the documents. The incorrect sentence segmentation
Table 11: Performance of our system for span, type and realis classification on KBP 2017 evaluation dataset [results released by the organizers]

| Runs | Span-P | Span-R | Span-F1 | Type-P | Type-R | Type-F1 | Realis-P | Realis-R | Realis-F1 | All-P | All-R | All-F1 |
|------|--------|--------|---------|--------|--------|---------|----------|---------|-----------|-------|-------|-------|
| I    | 58.95  | 56.53  | 57.72   | 45.21  | 39.53  | 43.30   | 41.60    | 37.16   | 39.87     | 32.64 | 31.31 | 31.96 |
| II   | 64.22  | 50.45  | 56.50   | 50.32  | 44.28  | 47.30   | 42.07    | 40.94   | 33.35     | 36.30 | 28.52 | 31.94 |
| III  | 57.44  | 54.44  | 55.90   | 45.88  | 43.48  | 44.65   | 42.07    | 39.87   | 33.35     | 32.60 | 31.30 | 31.90 |

Table 12: Performance comparison of our system on coreference resolution [results released by the organizers]

| Runs | B  | CeafE | MUC | BLANC | CoNLL |
|------|----|-------|-----|-------|-------|
| I    | 35.0 | 33.95 | 18.66 | 16.54 | 26.29 |
| II   | 34.34 | 33.63 | 22.90 | 17.94 | 27.20 |
| III  | 35.03 | 34.67 | 18.68 | 16.47 | 26.21 |

5.2 Conclusion and Future work

In this paper, we described TAMU’s participation in TAC KBP 2017 event nugget and coreference track. Our feature based system showed the advantage of using dependency parse tree based features for this task. Empirically, we also found that the joint modeling of event span detection and realis status identification helps in improving the performance. This is particularly interesting and we plan to continue our work in this direction.

References

Fei Cheng and Yusuke Miyao. 2017. Classifying temporal relations by bidirectional lstm over dependency paths. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers).

François Chollet. 2015. Keras. https://github.com/fchollet/keras.

Prafulla Kumar Choubey and Ruihong Huang. 2017a. Event coreference resolution by iteratively unfolding inter-dependencies among events. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pages 2114–2123.

Prafulla Kumar Choubey and Ruihong Huang. 2017b. A sequential model for classifying temporal relations between intra-sentence events. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pages 1797–1803.

Prafulla Kumar Choubey and Shubham Pateria. 2016. Garuda & bhasha at semeval-2016 task 11: Complex word identification using aggregated learning models. Proceedings of SemEval pages 1006–1010.

Pádraig Cunningham, John Carney, and Saji Jacob. 2000. Stability problems with artificial neural networks and the ensemble solution. Artificial Intelligence in medicine 20(3):217–225.
Greg Dubbin, Archna Bhatia, Bonnie Dorr, Adam Dalton, Kristy Hollingshead, Suriya Kandaswamy, Ian Perera, and Jena D. Hwang. 2016. Improving discern with deep learning.

James Ferguson, Colin Lockard, Natalie Hawkins, Stephen Soderland, Hamanesh Hajishirizi, and Daniel S. Weld. 2016. University of washington tac-kbp 2016 system description.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In Proceedings of the 43rd annual meeting on association for computational linguistics. Association for Computational Linguistics, pages 363–370.

Yu Hong, Yingying Qiu, Zengzhuan Xu, Wenxuanand Tang Jian Zhou, Xiaobin Wang, Liang Yao, and Jianmin Yao. 2016. Soochownlp system description for 2016 kbp slot filling and nugget detection tasks. In Proceedings of Ninth Text Analysis Conference.

Ruihong Huang, Ignacio Cases, Dan Jurafsky, Cleo Condoravdi, and Ellen Rifoff. 2016. Distinguishing past, on-going, and future events: The eventstatus corpus. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing.

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford’s multi-pass sieve coreference resolution system at the conll-2011 shared task. In Conference on Natural Language Learning (CoNLL) Shared Task.

Zhengzhong Liu, Jun Araki, Teruko Mitamura, and Eduard Hovy. 2016. Cmu-li at kbp 2016 event nugget track. In Proceedings of Ninth Text Analysis Conference.

Jing Lu and Vincent Ng. 2016. Utds event nugget detection and coreference system at kbp 2016. In Proceedings of the Ninth Text Analysis Conference.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In Association for Computational Linguistics (ACL) System Demonstrations. pages 55–60.

Todor Mihaylov and Anette Frank. 2016. Aiphes-hd system at tac kbp 2016: Neural event trigger detection and event type and realis disambiguation with word embeddings. In Proceedings of the TAC Knowledge Base Population (KBP) 2016.

Teruko Mitamura, Zhengzhong Liu, and Eduard Hovy. 2016. Overview of tac-kbp 2016 event nugget track. In Proceedings of Ninth Text Analysis Conference.

Thien Huu Nguyen, Adam Meyers, and Ralph Grishman. 2016. New york university 2016 system for kbp event nugget: A deep learning approach. In Proceedings of Ninth Text Analysis Conference.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In EMNLP. volume 14, pages 1532–1543.

Marta Recasens, Marie-Catherine de Marneffe, and Christopher Potts. 2013. The life and death of discourse entities: Identifying singleton mentions. In North American Association for Computational Linguistics (NAACL).

Taneeya Satyapanich and Tim Finin. 2016. Event nugget detection task: Umnbcs system.

K. Toutanova, D. Klein, C. Manning, and Y. Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In Proceedings of HLT-NAACL 2003.

Chen-Tse Tsai, Stephen Mayhew, Haoruo Peng, Mark Sammons, Bhargav Mangipudi, Pavankumar Reddy, and Dan Roth. 2016. Illinois ccg entity discovery and linking, event nugget detection and coreference, and slot filler validation systems for tac 2016.

Shang Chun Sam Wei, Igor Korostil, and Ben Hachey. 2016. Overview of sydney system for tac kbp 2015 event nugget detection.

Bishan Yang, Ndpandula Nakashole, Kisiel, Emmanouil A Platanios, Abulhair Saparov, Shashank Srivastava, Derry Wijaya, and Tom Mitchell. 2016. Cmuml micro-reader system for kbp 2016 cold start slot filling, event nugget detection, and event argument linking. In Proceedings of Ninth Text Analysis Conference.

Wenlin Yao, Saipravallika Nettyam, and Ruihong Huang. 2017. A weakly supervised approach to train temporal relation classifiers and acquire regular event pairs simultaneously. arXiv preprint arXiv:1707.09410.

Dian Yu, Xiaomai Pan, Boliang Zhang, Lifu Huang, Di Lu, Spencer Whitehead, and Heng Ji. 2016. Rpi blender tac-kbp2016 system description.

Ying Zeng, Bingfeng Luo, Yansong Feng, and Dongyan Zhao. 2016. Wip event detection system at tac kbp 2016 event nugget track. In Proceedings of TAC KBP 2016 Workshop, National Institute of Standards and Technology.