University of Washington TAC-KBP 2016 System Description

James Ferguson*, Colin Lockard*, Natalie Hawkins, Stephen Soderland, Hannaneh Hajishirzi, and Daniel S. Weld
Department of Computer Science & Engineering
University of Washington
Seattle, WA 98195
{jfferg, lockardc, nhawkins, soderlan, hannaneh, weld}@cs.washington.edu

Abstract

This document describes the University of Washington's event extraction system used in the Event Argument Extraction and Linking and Event Nugget Detection tasks of the 2016 TAC KBP competition. This system was composed of three components: Evento, a CRF-based extractor, NomEvent, which makes use a lexicon to build features to identify nominal triggers, and NewsSpike, which uses an unsupervised training process to produce a high-precision extractor (Zhang et al., 2015). These three methods combine to form a complementary system which performs better than any single individual component.

1 Overview

The Text Analysis Conference Knowledge Base Population (TAC-KBP) evaluation provides an opportunity to compare the performance of modern information extraction systems. The University of Washington (UW) participated in two tasks at the 2016 TAC-KBP: Event Argument Extraction and Linking (EAL) and Event Nugget Detection. This was the first year that UW participated in these tasks, using a system which included two completely new components: Evento and NomEvent.

2 System Overview

The UW event extraction system is composed of three separate systems which can each operate as independent event and argument extractors. Two of these systems were newly developed for the 2016 TAC KBP; the third, NewsSpike, is an existing UW event extractor.

Evento is a CRF-based structured event and argument extractor. It takes a pipelined approach in which each stage of the pipeline uses a loss-augmented training function allowing it to be tuned to improve either precision or recall.

NomEvent is a supervised extractor with a focus on extracting events triggered by nouns. It uses a lexicon of likely nominal event triggers generated through an automated process (described below) to generate features.

NewsSpike is trained using an unsupervised process based upon OpenIE principles, so the set of events it extracts is not based upon the Rich ERE (RERE) ontology (Zhang et al., 2015). In order to participate in the TAC KBP evaluation, a mapping was created from NewsSpike events to RERE. Only a subset of NewsSpike's events could be mapped to RERE events, so NewsSpike served as a low recall but high precision contributor to the overall system.

Both Evento and NomEvent use a pipelined approach, in which a document is passed through the following process: (1) Preprocessing with Stanford CoreNLP (POS, NER, dependency parsing, and lemmatization), (2) Entity Extraction, (3) Trigger Extraction and Classification, (4) Argument Classification, (5) Realis Classification.

The preprocessing step is identical for both systems, but they differ in the remaining steps. Both systems use linear classifiers to perform trigger and argument classification, but differ in the features used in their classifiers, as shown in Table

*Both authors contributed equally to the paper.

1. Evento and NomEvent were both trained on the ACE 2005 corpus, while NomEvent's training data was also supplemented with Rich ERE data from LDC2016E60.

2.1 Evento

Evento is a supervised system that uses a structured model with features primarily based on those used by (Li et al., 2013), which is the current state-of-the-art for models with discrete features.

2.1.1 Entity Extraction

Entity extraction in Evento uses a semi-Markov conditional random field (Sarawagi and Cohen, 2004). Given a sentence $x = (x_1, \ldots, x_n)$ the model considers sequences of labeled spans $\bar{s} = ((\ell_1, b_1, e_1), (\ell_2, b_2, e_2), \ldots, (\ell_k, b_k, e_k))$, where $\ell_i \in \{\text{Entity}, \text{Non-Entity}\}$ is a label for each span and $b_i, e_i \in \{0, 1 \ldots n\}$ are fenceposts for each span such that $b_i < e_i$ and $e_i = b_{i+1}$. The model places distributions over these sequences given the sentence as follows:

$$p_\theta(\bar{s}|x) \propto \exp\left(\theta^\top \sum_{i=1}^{k} f(x, (\ell_i, b_i, e_i))\right) \quad (1)$$

where f is a feature function that computes features for a span given the input sentence. The feature function we use includes both the union of token level features fired for each token in a span as well as features fired for the overall span. The specific features we use are outlined in Table 2.

We train this model on the gold entity annotations found in the ACE 2005 corpus[2]. In order to train the model, we maximize the conditional log likelihood of the training data augmented with a loss function via softmax-margin (Gimpel and Smith, 2010). We optimize using the AdaGrad algorithm of (Duchi et al., 2011) with L_2 regularization.

2.1.2 Evento Trigger Extractor

Both the trigger and argument classification stages in Evento are performed using linear-chain conditional random fields (CRF). Similar to entity extraction, we train the models by maximizing the conditional log likelihood of the training data augmented with a loss function and optimize using AdaGrad with L_2 regularization. During trigger extraction, each token in a sentence is assigned a label. Each label is either an event type we are interested in or *NO-EVENT* signifying that the token is not a trigger. The features we use for trigger classification are given in Table 1.

2.1.3 Evento Argument Extractor

As mentioned in the previous section, we use a CRF to perform argument classification. For every trigger identified in the previous step, the system assigns argument roles to each entity in the sentence. The possible roles depend on what arguments a particular event can take, as well as *NO-ARGUMENT* signifying that the entity did not participate in the event. Note that multiple triggers can occur in a sentence, so the system may have to classify an entity multiple times for separate event triggers. The features we used are outlined in Table 3.

2.2 NomEvent

The motivation behind NomEvent is to use existing NLP resources to develop an event extraction system focused on identifying events triggered by nouns.

We first aim to develop a lexicon of likely nominal event triggers by starting with a seed verb corresponding to an event and then searching WordNet and FrameNet for related nominal forms. This lexicon is then used to build features for a supervised classifier. A pre-trained Google Word2Vec model trained on Google News data[1] (Mikolov et al., 2013a; Mikolov et al., 2013b) was used in developing the lexicon and in the classifier.

NomEvent was developed with a focus on detecting events triggered by nouns, but the process was adapted detect events triggered by verbs as well. Both a nominal-trigger-only NomEvent system and an all-trigger NomEvent system were used in the evaluation, as described below in the descriptions of each run.

2.2.1 Lexicon Construction

In order to develop the lexicon of potential nominal event triggers, we start with a seed verb for each event in the ontology. In most cases, the event

[2]https://www.ldc.upenn.edu/collaborations/past-projects/ace

[1]https://code.google.com/archive/p/word2vec/

Trigger Features	
Evento & NomEvent	Token bigram Dependency bigram Dependent lemma Governor lemma NER types in sentence Entity types in sentence POS tags
Evento only	Basic WordNet Synonyms Brown Clusters
NomEvent only	Token Word2Vec embedding Dependency path to sentence nouns Document-level Event Basket hits Event Basket Bag of Words Event Basket Distance Comparison WordNet lexname WordNet traversal features

Table 1: A comparison of features used in the trigger classifiers of Evento and NomEvent

Evento Entity Features	
Word Features	Span Features
Word properties	Token n-gram context
Token	Span length
Prefixes	Dependency arcs entering/leaving span
Suffixes	Phrase type of span in constituency tree

Table 2: Features used during the entity extraction step of Evento. Word properties capture information about the capitalization, numbers, and punctuation in a word.

subtype is used as the seed verb, such as "attack" for Conflict.Attack or "meet" for Contact.Meet. In cases where the event subtype was not suitable as a single verb, such as for Personnel.Start-Position, a human user selected a word to use as a seed (in this example, "hire"). For a few events, there was not a single obvious word that completely characterized the event, such as Personnel.EndPosition; potential verbs for this event might include "quit", "fire", or "layoff". For the system presented here, one of these words was selected by the user (in this example, "quit").

Once a seed verb has been selected for each event, a human user searched WordNet (Fellbaum, 1998) for synsets in which that word participated and selected one or more synsets with definitions that best characterized that event. This operation took less than 5 minutes of user time per event. An alternate version was explored where the most common synset for each seed verb was used, but this method was found to produce less accurate results.

Once a selection of synsets is made, all lemmas for each synset were retrieved, along with immediate hyponyms. For each lemma on the resulting list, the cosine distance between the Word2Vec embeddings for that lemma and its corresponding seed verb was calculated, and any lemma with a distance greater than 0.75 was discarded. For each remaining lemma, all derivationally-related nouns are then added to the lexicon. (For the version of the system which identifies triggers from any part of speech, all related forms are added). Table 4 shows the seed

Evento Argument Features
Token bigrams
POS bigrams
Distance to trigger word
Dependency path to trigger word
NER Tags

Table 3: Features used during the argument extraction step of Evento.

NomEvent Lexicon Generation	
Event	Conflict.Attack
Seed verb	attack
Hand-selected synsets	attack.n.01 attack.v.01 attack.v.06
Resulting Lexicon (selections)	Occupation Offensive Onrush Raid Storm Strike Torpedo

Table 4: The seed verb and synsets used on the creation of the NomEvent lexicon entry for Conflict.Attack are shown, along with some of the twenty-six words in the resulting lexicon.

verb and synsets used for the Conflict.Attack event, as well as selected nouns in the resulting lexicon.

When a lemma is added to the lexicon, an additional set of features related to that lemma are also saved for use in the trigger classifier. These features, listed on Table 1 as "WordNet traversal features", include the cosine distance to the seed verb, the total number of times that lemma appeared in the WordNet traversal for that seed verb, the WordNet corpus frequency for that lemma, and the percent of all corpus mentions for that synset which that lemma represented.

In addition, a FrameNet search is made for each seed verb (Baker et al., 1998). If a frame is found which matches the event, any nouns participating in that frame are added to the lexicon if not already present.

2.2.2 Entity Extraction

A CRF-based entity extractor was trained on the ACE 2005 corpus. Features included part-of-speech, NER tag, and word shape.

2.2.3 NomEvent Trigger Extractor

The NomEvent trigger extractor uses an L2-regularized multilabel logistic regression classifier to process each word in a sentence in sequence and apply a label (either an event subtype or "None"). In addition to conventional features (summarized on Table 1), several features are calculated based on the lexicon. We use "Event Basket" to designate the words in the lexicon associated with each event.

The first lexicon-based feature is the number of words in the document which are present in each Event Basket. This feature is based on the observation that many newswire documents constitute a narrative which repeatedly references elements of an event throughout the article, so potential triggers which appear in a surrounding context containing many words related to that event are highly likely to relate to an event (Huang and Riloff, 2011).

The second lexicon-based feature is a simple bi-

nary vector indicating if the token lemma matched any of the lexicon words. If one of the words was matched, the features described above that were gathered in the traversal of WordNet are included as well.

The final lexicon-based feature is, for each event, the average cosine distance between the word embedding for the token lemma and the embedding for each word in the event basket for that event. This provides a score for each event which represents the distance of the token to the set of embeddings which represent that event.

2.2.4 NomEvent Argument Extractor

When a trigger has been identified in a sentence, all entities in that sentence are classified to determine whether they are argument for the event. The argument classifier is identical to the trigger classifier, but with the addition of several features: Dependency path to trigger, dependency path length, distance to trigger, entity type, NER type of previous and next word, and whether the prior or next lemma were on a small hand-collected list of words such as "the", "to", and "of". In addition, the event basket Word2Vec similarity comparison was replaced with Word2Vec comparisons with a small manually generated list of words related to event roles such as "city", "company", and "attacker".

2.3 Adapting NewsSpike to TAC Ontology

NewsSpike is an event extractor which uses an unsupervised method based on Open Information Extraction principles to generate and cluster data on which it is trained (Zhang et al., 2015). The set of events on which it is trained is thus discovered from data and does not correspond to a pre-set ontology. In order to adapt NewsSpike to the Rich ERE ontology, a mapping was created to map each of the 150 events which NewsSpike extracts to an RERE event, or to null if no RERE event existed. This mapping was completed manually. Of the 150 NewsSpike event, only 65 could be mapped to the events in the ontology for this year's EAL and Nugget evaluations. Many NewsSpike events are fine-grained, so of these 65, many only partially corresponded to the ERE counterpart; for example, the NewsSpike events "apologize", "assure", "congratulate", "reach out", "talk", and "warn" were all mapped to Contact.Correspondence. The existing NewsSpike system was trained on a wide ranging corpus of news articles scraped from the web. A version of NewsSpike trained on a domain better corresponding to the TAC ontology would likely provide more meaningful results.

3 Description of System Runs

3.1 Event and Argument Extraction and Linking

UW submitted five runs for the English EAL task. In this task, teams were presented with a corpus of 30,002 documents, evenly divided between newswire and discussion forum text, with the goal of extracting arguments participating in a set of 18 event subtypes, identifying both event subtype and role. Both Evento and NomEvent were trained on the ACE 2005 corpus, with NomEvent's training corpus supplemented with Rich ERE data from LDC2016E60.

Run Washington1 aimed for maximum recall by combining the Evento and NewsSpike systems with the NomEvent system trained to classify all parts of speech. The union of the events returned by the three systems was used, with the Evento result chosen when overlapping argument extractions disagreed on the extent or role of the argument.

Run Washington2 was identical to Washington1 but substituted the NomEvent system trained to only extract nominal events.

Run Washington3 consisted solely of the Evento system.

Run Washington4 consisted solely of the NomEvent system, classifying all potential triggers.

The final run, Washington5, aimed for a high-precision result by considering the results returned by the Evento, NewsSpike, and NomEvent (all-part-of-speech) systems and keeping only results returned by at least two of the three systems.

3.2 Event Nugget Identification

The 2016 English Event Nugget task provided a corpus of 169 documents, split between newswire and discussion forum text and required teams to extract event triggers corresponding to the same ontology of 18 events used in EAL. UW submitted three runs for this task.

The first run, Washington1, consisted of the union of an Evento run tuned for F1, a NomEvent run trained on all parts of speech, and a NewsSpike run. Run Washington2 consisted of the union of an Evento run tuned for F1, a NomEvent run trained on nominal events, and a NewsSpike run. Washington3 was identical to Washington1 but was tuned for high precision.

4 Results

Table 5 shows detailed results of the EAL evaluation, in which UW scored above the median for both the Argument and Linking scores. The median Argument score over the top performing system from each team was 3.0; Washington4 topped this with a 3.3, as did Washington1 with a 3.2. For the linking score, Washington1 posted a 2.0, above the median of 1.6. As anticipated, Washington1 posted the highest recall of the UW systems, while Washington5 posted the highest precision.

Table 6 shows detailed results of the event nugget evaluation. Washington1 posted higher recall and F1 scores than the other UW systems, while Washington3 turned in the highest precision, as expected. An examination of the event breakdown revealed that Washington1 earned its highest F1 scores on the Life.Injure (0.64), Life.Die (0.54), and Justice.ArrestJail (0.60) events, while it struggled on Contact.Contact (0.01), Contact.Broadcast (0.01), and Transaction.Transaction (0.0), Manufacture.Artifact (0.0) and Movement.TransportArtifact (0.0).

Failure Analysis The poor performance on some events is likely due to the distribution of events in the test data as compared to the ACE 2005 corpus which provided the bulk of the training data for the Evento and NomEvent systems, as shown in Figure 1. These five events made up almost 20% of events in this year's Event Nugget evaluation, so training on more representative data would likely have significantly boosted the performance of our systems.

Evento places a lot of weight on the large number of lexical features it uses. Because of this, it generalizes poorly to triggers that do not appear in the training set, especially when they appear with new contexts that also were not seen in training. This results in low recall for events with a diverse set of triggers, such as *start-org*, *transfer-ownership*, and *transfer-money*.

A failure analysis of NomEvent discovered that 58% of false positives corresponded to a misclassification of closely related events, such as classifying a Contact.Broadcast event as Contact.Correspondence. The vast majority of missed triggers (88%) were words which were not in the generated lexicon.

5 Related Work

There have been a number of approaches utilizing pipelines for event extraction in the past (Liao and Grishman, 2010; Hong et al., 2011). Recent work has also explored joining various steps together, such as event and argument identification using a structured perceptron (Li et al., 2013), or doing joint inference over entities, triggers, and arguments using an ILP (Yang and Mitchell, 2016). More recently there has been a shift towards deep learning approaches to the problem (Chen et al., 2015; Nguyen et al., 2016).

A lexicon of potential nominal event triggers was developed in (Do et al., 2011), which used WordNet to gather all derivationally related nouns for all synsets of a large corpus of seed verbs, as well as gathering nouns from FrameNet. Do et al. then pruned this list using a rough set of heuristics, such as edit distance to event verb, whereas our work starts with a more targeted list and prunes using Word2Vec similarity. Their focus was on minimally supervised event causality detection rather than argument extraction, so they did not use this list to build features for a supervised classifier. Trigger classifiers using features related to WordNet morphological connection have also previously been explored in (Ahn, 2006), which used synset ID as a feature.

Information Extraction focused on nouns has also been conducted in the context of relation extraction. ReNoun (Yahya et al., 2014) introduced an Open IE system which started with a set of seed patterns to learn dependency patterns of nominal relations from a large corpus. Nominal lexicons such as Nomlex (Macleod et al., 1998) have also been used as features in event extraction systems such as (Li et al., 2013), in that case as a source of base lemmas. Our

TAC KBP 2016 EAL Evaluation Results								
System	TP	FP	FN	ArgP	ArgR	ArgF1	ArgScore	LinkScore
Washington1	440	1223	6065	26.5	**6.8**	**10.8**	3.2	**2.0**
Washington2	343	948	6162	26.6	5.3	8.8	2.6	1.3
Washington3	247	717	6258	25.6	3.8	6.6	2.0	0.7
Washington4	327	691	6178	32.1	5.0	8.7	**3.3**	1.5
Washington5	120	144	6385	**45.5**	1.8	3.5	1.6	0.3

Table 5: Scores for the Event Argument Extraction and Linking evaluation.

TAC KBP 2016 Event Nugget Evaluation Results							
		Micro			Macro		
System	Attributes	Prec	Rec	F1	Prec	Rec	F1
Washington1	plain	50.19	**35.02**	**41.25**	47.34	**33.11**	**38.97**
	mention_type	42.15	**29.41**	**34.65**	38.95	**27.50**	**32.24**
	realis_status	36.20	**25.25**	**29.75**	34.18	**23.58**	**27.91**
	mention_type+realis_status	30.71	**21.42**	**25.24**	28.35	**19.75**	**23.28**
Washington2	plain	49.76	33.01	39.69	47.14	31.09	37.47
	mention_type	41.83	27.75	33.36	38.68	25.79	30.95
	realis_status	36.38	24.13	29.02	34.61	22.66	27.39
	mention_type+realis_status	30.97	20.55	24.70	28.78	19.04	22.92
Washington3	plain	**62.15**	26.64	37.29	**57.12**	24.54	34.33
	mention_type	**55.96**	23.99	33.58	**50.89**	21.97	30.69
	realis_status	**45.22**	19.38	27.14	**40.75**	17.66	24.64
	mention_type+realis_status	**41.10**	17.62	24.66	**36.61**	15.92	22.19

Table 6: Results for the 2016 Event Nugget Detection evaluation.

testing on a subset of five events showed that nouns in Nomlex appeared as event triggers 35% less frequently than words in our lexicon.

6 Conclusions

We participated in the TAC KBP 2016 EAL and Event Nugget evaluations with a series of entries combining three UW event extraction systems, Evento, NomEvent, and NewsSpike. Our entries achieved results demonstrating that the systems are complementary, and while they were not competitive with the top entrant, they did place in the top half of entries to the EAL task.

Acknowledgments

This research was supported in part by ONR grant N00014-11-1-0294, DARPA contract FA8750-13-2-0019, NSF (IIS 1616112), and ARO grant W911NF-13-1-0246.

References

Ahn, D. (2006). The stages of event extraction.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The berkeley framenet project. In *COLING-ACL*.

Chen, Y., Xu, L., Liu, K., Zeng, D., and Zhao, J. (2015). Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of the Association for Computational Linguistics*.

Do, Q., Chan, Y. S., and Roth, D. (2011). Minimally supervised event causality identification. In *EMNLP*.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books.

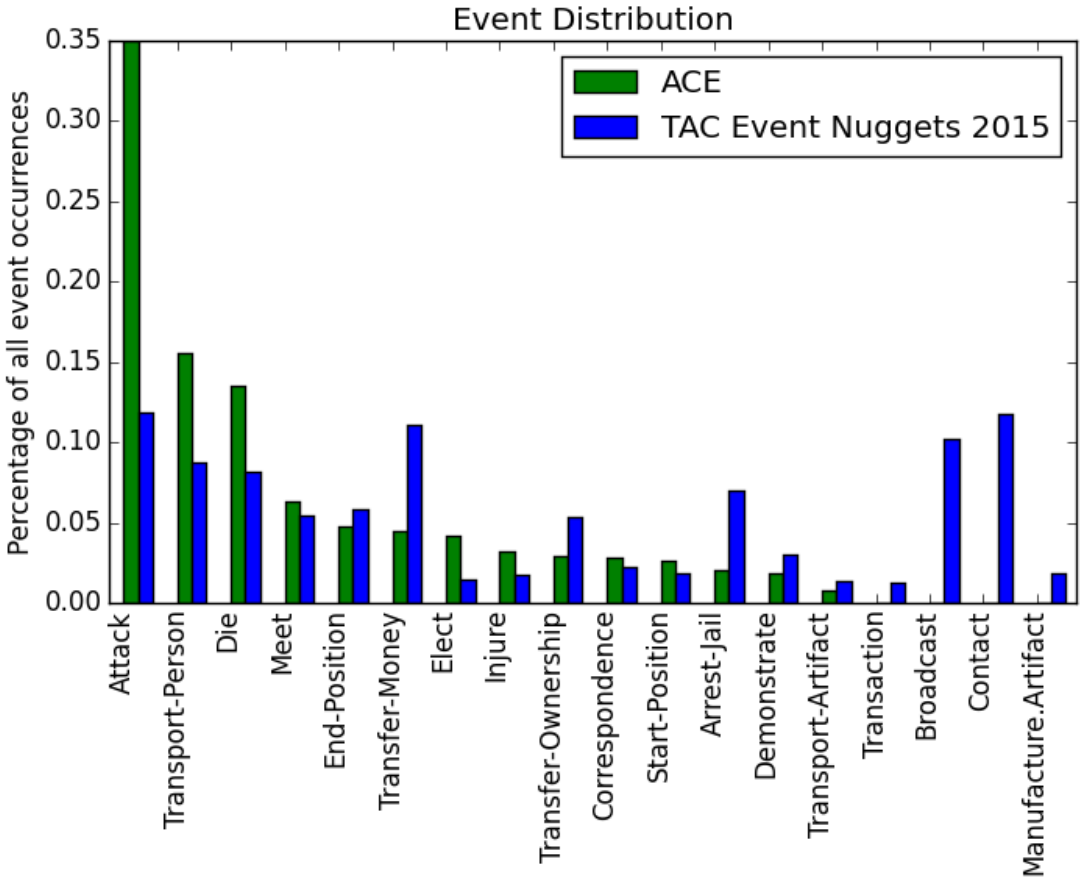

Figure 1: Comparison of event distribution in ACE 2005 dataset as compared to the TAC KBP Event Nugget 2015 task.

Gimpel, K. and Smith, N. A. (2010). Softmax-Margin CRFs: Training Log-Linear Models with Cost Functions. In *Proceedings of the North American Chapter for the Association for Computational Linguistics*.

Hong, Y., Zhang, J., Ma, B., Yao, J.-M., Zhou, G., and Zhu, Q. (2011). Using cross-entity inference to improve event extraction. In *Proceedings of the Association for Computational Linguistics*.

Huang, R. and Riloff, E. (2011). Peeling back the layers: Detecting event role fillers in secondary contexts. In *ACL*.

Li, Q., Ji, H., and Huang, L. (2013). Joint event extraction via structured prediction with global features. In *ACL*.

Liao, S. and Grishman, R. (2010). Using document level cross-event inference to improve event extraction. In *Proceedings of the Association for Computational Linguistics*.

Macleod, C., Grishman, R., Meyers, A., Barrett, L., and Reeves, R. (1998). Nomlex: A lexicon of nominalizations. In *Proceedings of EURALEX*, volume 98, pages 187–193. Citeseer.

Mikolov, T., Chen, K., Corrado, G. S., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.

Nguyen, T. H., Cho, K., and Grishman, R. (2016). Joint event extraction via recurrent neural networks. In *Proceedings of the North American Chapter for the Association for Computational Linguistics*.

Sarawagi, S. and Cohen, W. W. (2004). Semi-Markov Conditional Random Fields for Information Extraction. In *Proceedings of Advances in Neural Information Processing Systems*.

Yahya, M., Whang, S. E., Gupta, R., and Halevy, A. Y.

(2014). Renoun: Fact extraction for nominal attributes. In *EMNLP*.

Yang, B. and Mitchell, T. (2016). Joint Extraction of Events and Entities Within a Document Context. In *Proceedings of the North American Chapter for the Association for Computational Linguistics*.

Zhang, C., Soderland, S., and Weld, D. S. (2015). Exploiting parallel news streams for unsupervised event extraction. *TACL*, 3:117–129.