Illinois CCG Entity Discovery and Linking, Event Nugget Detection and Co-reference, and Slot Filler Validation Systems for TAC 2016

Chen-Tse Tsai, Stephen Mayhew, Haoruo Peng,
Mark Sammons, Bhargav Mangipundi, Pavankumar Reddy, and Dan Roth
University of Illinois, Urbana-Champaign, USA
{ctsai12,mayhew2,hpeng7,mssammon,mangipu2,
muddire2,danr}@illinois.edu

Abstract

The University of Illinois CCG team participated in three TAC 2016 tasks: Entity Discovery and Linking (EDL); Event Nugget Detection and Co-reference (ENDC); and Slot Filler Validation (SFV). The EDL system includes Spanish and Chinese named entity recognition, cross-lingual wikification, and nominal head detection. The ENDC system identifies event nugget mentions and puts them into co-reference chains. We develop ENDC based on English and it works on Spanish and Chinese through translations. The SFV system uses a set of classifiers, one per target relation, trained with the gold assessed TAC Cold Start Knowledge Base Population responses, filtered using performance on this data.

1 Tri-Lingual Entity Discovery and Linking

1.1 Overview

We focus on the Chinese and Spanish EDL subtasks this year. Last year, we developed a strong translation-based Spanish EDL system, which translates Spanish documents into English and then applies Illinois English NER (Ratinov and Roth, 2009) and Wikifier (Ratinov et al., 2011) on the translated text. Although this system achieved the best results on the Spanish sub-task, we decided to take another approach this year: building a monolingual system for each language. The main reasons are two-folds. First, there are much more test documents this year, which makes using Google Translation expensive. Second, we have more experience with the monolingual and cross-lingual models since last year's first attempt of working on Spanish. It would be interesting to compare the translation-based system with monolingual systems.

The pipeline of our system is shown in Figure 1. The top part of the figure solves the named entity discovery and linking problem, namely, the goal of last year's task. The bottom part handles the new problem introduced this year: nominal head detection and the co-reference problem between nominal mentions and named entity mentions. Each component in the pipeline is described in detail in the following sections.

1.2 Spanish and Chinese NER

The first step in the pipeline is recognizing named entity mentions. We build a Spanish and a Chinese NER model separately. The models are based on our cross-lingual NER model (Tsai et al., 2016). Although the key idea in Tsai et al. (2016) is that a cross-lingual wikifier can generate good language-independent NER features, thus a model trained on one language can be applied on the text of another language, we also show that the newly proposed wikifier features are very useful in monolingual models (in which the training and test documents are in the same language).

We use last year's training and evaluation documents and the ERE datasets[1] to construct training data for each language. Note that the Spanish model is only trained on the Spanish training data and the Chinese model is only trained on the Chinese training data, therefore the models are monolingual. We use Stanford Spanish tokenizer to preprocess Spanish documents, whereas our Chinese model is character-based, that is, each Chinese

[1] LDC2015E107,LDC2015E112, and LDC2015E78

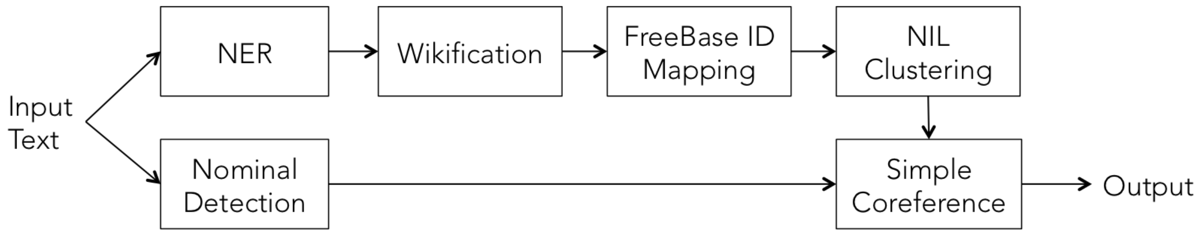

Figure 1: System pipeline

character is a token. In our experiments, applying Stanford Chinese word segmenter actually degrades the performance. We also gather gazetteers for Spanish and Chinese from Wikipedia titles, and train brown clusters using Wikipedia articles for each language.

1.3 Entity Linking

After extracting named entity mentions, the next step is to grounding these mentions to Wikipedia. In this step, we use the model proposed in Tsai and Roth (2016), which uses cross-lingual word and title embeddings to generate similarities between a foreign mention and English title candidates. Note that we only ground mentions to the intersection of the English and target language Wikipedia.

We then obtain the corresponding FreeBase ID using the links between Wikipedia titles and FreeBase entries if the mention is grounded to some Wikipedia entry.

Finally, we perform the NIL clustering algorithm that we developed last year (Sammons et al., 2015b) on the mentions which cannot be grounded to any FreeBase entry. In this algorithm, the similarity between mentions is based on the token-based Jaccard similarity of surface strings.

1.4 Nominal Head Detection

A new requirement this year is to extract heads of nominal nouns which refer to specific or individual entities. Since there is no training data specific to this definition provided, we take all nominal head annotations in the ERE datasets to be our training data. We then simply train our NER model on this data to produce nominal head detection models for Spanish and Chinese separately.

1.5 Simple Co-reference

In the final step, we try to link each nominal head mention to the named entity mentions, that is, resolving the co-reference problem between nominal nouns and named entities. We use the following simple heuristic rules. For each nominal head mention, we find the closest mention (either a nominal or a name) to the left which has the same entity type. If this closest mention is a nominal, the surface form is also required to be identical. We then add the current nominal mention to the cluster of the closest matching mention. Note that we also set a threshold on how far can we go to do this search. If no suitable mention is found within this window, the current nominal mention is discarded, since this nominal mention could refer to some generic noun rather than a specific entity.

Language	Precision	Recall	F1
Spanish	82.33	63.15	71.48
Chinese	64.98	44.74	52.99

Table 1: Performance of nominal head detection on the ERE datasets.

1.6 Evaluation

Since there is no end-to-end development document this year, we evaluate the named entity discovery and linking component and the nominal head detection component separately during developing our systems. We first show the numbers we got during development, and the official evaluation results are listed in the last section.

1.6.1 Nominal Head Detection

We take the ERE datasets and randomly make 80% of the data training/development documents and the rest 20% are the test documents. The performance of our nominal head detection models are listed in Table 1. We can see that the performance of Spanish is much better than Chinese. One possible reason is that most of the mentions in Spanish have only single token, whereas Chinese mentions are usually two characters long (We also use a character-based model here).

1.6.2 Spanish and Chinese EDL

We evaluate the named entity discovery and linking component on TAC 2015 evaluation documents. Table 2 shows the F1 scores of three different metrics. Comparing to the best systems last year, our approach achieves much better NER scores (strong typed mention match) on both Spanish and Chinese. However, when considering FreeBases ID in evaluation (strong typed all match), we are only slightly better on Spanish but 0.8 points worse than the best Chinese system last year. We suspect this is due to the conversion between Wikipedia titles and FreeBase IDs. Besides the links in the FreeBase dump, we also utilized the FreeBase search API last year, to convert Wikipedia titles to FreeBase IDs. However, since FreeBase API is closed this year, we fail to map some Wikipedia titles to the corresponding FreeBase entries.

The top Spanish system of 2015 is our translation-based system, which uses Google Translate to translate Spanish documents into English, and then applies Illinois NER and Illinois Wikifier on the translated English text. It is not surprising that our monolingual system this year outperforms the translation-based system, since there is in-domain Spanish training data for NER. Nevertheless, the translation-based system is still very interesting as it only uses the default English models for NER and Wikification, that is, the models were not re-trained. We only trained an entity type classifier based on FreeBase types to assign one of the five entity types to the extracted mentions, since the default model of Illinois NER uses CoNLL labeling scheme.

Another interesting observation is that character-based Chinese NER is better than a word-based model which pre-tokenizes text using a Chinese word segmenter. Moreover, our model (Illinois NER) which was designed for English NER works well on Chinese.

1.6.3 Official Evaluation Results

We present the official evaluation results in this section. Table 3 shows our performance on entity discovery and linking, and Table 4 lists the results of nominal head detection.

For EDL, comparing to the development performance (Table 2), we achieve similar level of performance. However, for the nominal head detection, we are much worse than what we got on the development data (Table 1). One possible reason is due to different definitions of annotations. All nominal nouns in the text are annotated in the ERE dataset. In contrast, only specific and individual nominal nouns are annotated in the evaluation documents. Nevertheless, it does not explain the poor recall if the annotations in evaluation is a subset of annotations in the ERE data. More analysis of the poor performance on nominal head detection is required.

Measure	2015 Top	Our Approach
Spanish		
strong mention match	78.7	**79.8**
strong typed mention match	74.7	**77.9**
strong typed all match	69.2	**69.7**
Chinese		
strong mention match	79.9	**80.5**
strong typed mention match	76.9	**78.3**
strong typed all match	**72.2**	71.4

Table 2: Performance of Spanish and Chinese entity discovery and linking on the evaluation documents of 2015.

Measure	Pre.	Rec.	F1
Spanish			
strong mention match	88.9	78.6	83.4
strong typed mention match	85.6	75.7	80.4
strong typed all match	78.5	69.4	73.6
mention ceaf	85.0	75.2	79.8
Chinese			
strong mention match	87.8	75.6	81.2
strong typed mention match	83.0	71.5	76.8
strong typed all match	72.8	62.7	67.4
mention ceaf	79.0	68.0	73.1

Table 3: The official evaluation results of Spanish and Chinese entity discovery and linking.

2 Event Nugget Detection and Co-reference

In this section, we describe our submission to the TAC KBP event task. Our team participated in the TAC KBP Event Nugget (EN) track. It includes two multi-lingual sub-tasks: event nugget detection, event co-reference based on predicted event nuggets, both for English, Spanish and Chinese. We implement both supervised and dataless methods on these sub-tasks for English and support Spanish and Chinese via machine translation. The supervised method employs rich lexical and semantic features, while the dataless method mod-

Measure	Pre.	Rec.	F1
Spanish			
strong mention match	37.0	51.1	42.9
strong typed mention match	35.3	48.8	41.0
strong typed all match	12.5	17.2	14.5
mention ceaf	19.1	26.4	22.2
Chinese			
strong mention match	22.5	31.5	26.2
strong typed mention match	24.9	29.8	24.9
strong typed all match	6.9	9.7	8.1
mention ceaf	12.1	16.9	14.1

Table 4: The official evaluation results of Spanish and Chinese nominal head detection.

els the similarity between each event nugget pair or a nugget and an event type using semantic representations.

We describe our supervised and dataless event detection and co-reference techniques separately, and we show how we adapt our system to other languages.

2.1 Supervised Approach

Event Nugget Detection We use a stage wise classification approach to extract all events (Ahn, 2006; Chen and Ng, 2012) based on an extension of our system from last year's TAC KBP task (Sammons et al., 2015a). We first train a 34-class classifier (33 event sub-types and one non-event class) to detect event nuggets and classify them into different types. There are two different changes: 1) instead of applying it on each token, we apply this classifier on every *Semantic Role Labeling* (SRL) predicate to determine the event type; 2) during evaluation, instead of 18 types chosen by the task guideline.

Features for this supervised classifier includes lexical features, features from parser, *Named Entity Recognition* (NER), *Semantic Role Labeling* (SRL), entity co-reference and WordNet, and other semantic features from *Explicit Semantic Analysis* (ESA) (Gabrilovich and Markovitch, 2005; Gabrilovich and Markovitch, 2007) and Brown Clusters (Brown et al., 1992). More details can be found in our system description from last year (Sammons et al., 2015a).

We then apply a classifier using the same set of rich features on each detected event nugget to get *REALIS* information (ACTUAL, GENERIC or OTHER).

Event Co-reference We employ an event-pair model for event co-reference, which is similar to the mention-pair co-reference model (Denis and Baldridge, 2007) in entity co-reference. The similarity between event nugget pairs is trained based on a supervised model. We then employ a greedy clustering method to put every event nugget into co-reference chains. We first make a decision on each event nugget pair (whether they are linked or not) and then put all linked event nuggets into the same event co-reference chain.

Features for the supervised model can be put into four categories: 1) event nugget features: all features for event nugget detection and their conjunctions between two events nuggets; 2) event argument features: we get approximated event arguments directly through SRL. We first extract the sentence containing an event nugget, and then use SRL to extract SRL arguments. We treat them as event arguments. The deterministic mapping detail is provided in Peng et al. (2016). Though the event arguments we get are not precise, they are sufficient for event co-reference (supported by analysis in Peng et al. (2016)). We apply all event nugget detection features on the arguments and their conjunctions between arguments of two events nuggets; 3) event entity features: we get event entities directly through entity co-reference. We run entity co-reference on the whole document. Then, similar to the construction of event argument features, we extract sentences containing event nuggets, and then use entity mentions (as annotated by co-reference) in these sentences as event entities. We apply all event nugget detection features on the entities and their conjunctions between entities of two events nuggets; 4) pair-wise features: distance, ESA similarities of two events nuggets and number of co-referent entity mentions of two events nuggets.

Learning Model We choose *Support Vector Machine* (SVM) with L2 loss to train all the classifiers. We use Illinois NLP packages[2] for NER, SRL, and Entity Co-reference.

Domain Adaptation Apart from the KBP training data, we use ACE2005 as an additional source of our training data. The ACE event taxonomy is similar to that of the KBP task. To enable the domain adaptation from ACE to KBP, we employ the same techniques as described in (Sammons et al., 2015a) to enlarge the training data.

[2] `http://cogcomp.cs.illinois.edu/page/software_view/`

2.2 Dataless Approach

We also pursue an approach to understanding events that we believe to be more feasible and scalable. Fundamentally, event detection is about identifying whether an event in context is semantically related to a set of events of a specific type; and, event co-reference is about whether two event mentions are semantically similar enough to indicate that the author intends to refer to the same thing. Therefore, if we formulate event detection and co-reference as semantic relatedness problems, we can scale it to deal with a lot more types and, potentially, generalize across domains. Moreover, by doing so, we facilitate the use of a lot of data that is not part of the existing annotated event collections and not even from the same domain. The key challenges we need to address are those of how to represent events, and how to model event similarity; both are difficult partly since events have *structure*.

Our dataless approach builds on two key ideas. First, to represent event structures, we use the general purpose nominal and verbial semantic role labeling (SRL) representation. This allows us to develop a structured representation of an event. Second, we embed event components, while maintaining the structure, into multiple semantic spaces, induced at a contextual, topical, and syntactic levels. These semantic representations are induced from large amounts of text in a way that is completely independent of the tasks at hand, and are used to represent both event mentions and event types into which we classify our events. The combination of these semantic spaces, along with the structured vector representation of an event, allow us to directly determine whether a candidate event mention is a valid event or not and, if it is, of which type. Moreover, with the same representation, we can evaluate event similarities and decide whether two event mentions are co-referent. Consequently, the proposed MSEP (Minimally Supervised Event Pipeline), can also adapt to new domains without any training. Our dataless approach relies on even fewer resources than traditional unsupervised approaches. Table 5 summarizes the differences.

An overview of the system is shown in Figure 2. A few event examples are *all* the supervision MSEP needs; even the few decision thresholds needed to be set are determined on these examples, once and for all, and are used for *all* test cases we evaluate on. More details of the MSEP system can be referred to Peng et al. (2016).

	Supervised	Unsupervised	MSEP
Guideline	✓	✓	✓
In-domain Data	✓	✓	✗
Data Annotation	✓	✗	✗

Table 5: Comparing requirements of MSEP and other methods. Supervised methods need all three resources while MSEP only needs an annotation guideline (as event examples).

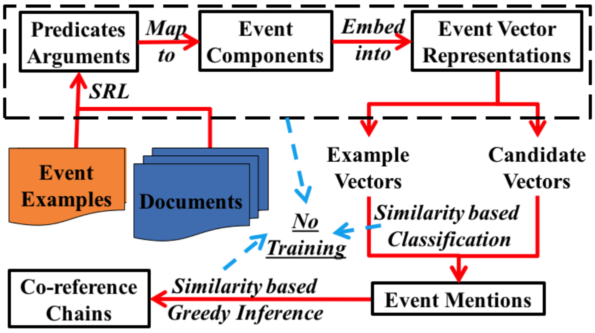

Figure 2: An overview of the end-to-end MSEP system. “Event Examples” are the only supervision here, which produce “Example Vectors”. No training is needed for MSEP.

2.3 Multi-lingual Setting

We develop our system (both supervised and dataless approaches) based on English. To support other languages, we directly translate the test documents from the target language to English via automatic machine translation[3].

After event detection and co-reference decisions have been made based on the translated text, we then map event nuggets back to source languages based on part-of-speech information. This process is carried out for both supervised and dataless approaches on Spanish and Chinese.

2.4 Evaluation

We select 50 documents from last year’s task test corpus as the development set. These selected documents contain genres of both news articles and discussion forums. For the KBP event nugget detection task, we submit the following two runs.

1. Trial One: supervised event detection as described in Section 2.1

[3] We use Google Translation here.

2. Trial Two: dataless event detection as described in Section 2.2

Results on English are shown in Table 6, while results for Spanish and Chines are shown in Table 7 and Table 8 respectively.[4] We observe that the system's performance is much lower compared to that of last year. A possible reason is that we tune the system's performance for the complete set of 34 event types while the evaluation is only carried out selected 18 types.

Table 6: Event Nugget Detection Test Results on English.

	Precision	Recall	F1
Trial One (Supervised)			
Detection	49.79	44.27	46.87
Type	42.61	37.26	39.76
Realis	35.53	31.40	33.34
Type+Realis	30.10	27.87	28.94
Trial Two (Dataless)			
Detection	47.23	42.66	44.83
Type	40.54	35.81	38.03
Realis	34.80	30.27	32.38
Type+Realis	29.13	26.79	27.91

Table 7: Event Nugget Detection Test Results on Spanish.

	Precision	Recall	F1
Trial One (Supervised)			
Detection	46.23	49.99	48.04
Type	35.79	37.86	36.80
Realis	34.69	37.77	36.16
Type+Realis	25.29	26.96	26.10
Trial Two (Dataless)			
Detection	45.31	48.30	46.76
Type	34.11	37.23	35.60
Realis	33.62	36.87	35.17
Type+Realis	24.74	25.88	25.30

For the KBP event nugget co-reference task, we submit the following three runs.[5]

1. Trial One: supervised event detection and co-reference as described in Section 2.1

Table 8: Event Nugget Detection Test Results on Chinese.

	Precision	Recall	F1
Trial One (Supervised)			
Detection	15.32	53.96	23.86
Type	14.63	49.85	22.62
Realis	11.69	40.26	18.12
Type+Realis	10.25	38.67	16.20
Trial Two (Dataless)			
Detection	13.43	53.66	21.48
Type	12.01	46.51	19.09
Realis	9.26	37.73	14.87
Type+Realis	8.69	35.65	13.97

2. Trial Two: dataless event detection and co-reference as described in Section 2.2

3. Trial Three: supervised event detection as described in Section 2.1 with dataless co-reference as described in Section 2.2

Results are shown in Table 9.[6] In this table, "AVG" stands for *CoNLL Average*, which is the average score of MUC, B^3, $CEAF_e$ and BLANC.

3 Slot Filler Validation

For the TAC 2016 SFV evaluation, the Illinois CCG SFV system was run on the English CSSF system outputs.

We treat the Slot Filler Validation (SFV) task as an entailment problem, with each Cold Start Slot Fill (CSSF) output used to generate a corresponding entailment pair. In order to match the standard entailment task, and to ensure that the techniques we develop are applicable to the entailment domain, we use no information about which system predicted a given response, nor do we combine multiple responses from different systems.

3.1 System Design

The Illinois CCG SFV system has a pipeline architecture, as shown in Figure 3. For each CSSF system output, the system first identifies the provenance document(s). If more than one document is specified, the query is ignored (i.e., left unfiltered) (see Section 3.2).

For the outputs that are not ignored, the provenance document is retrieved and cleaned up prior

[4] In the official evaluation, we left approximately a quarter of the test documents unprocessed due to a mistake. Here we report the corrected results.

[5] In the official evaluation, we only submitted the first two runs. The third run here is served as an ablation study.

[6] In the official evaluation, we left approximately a quarter of the test documents unprocessed due to a mistake. Here we report the corrected results.

Table 9: Event Nugget Co-reference Results on the Test Set.

	MUC	B^3	$CEAF_e$	BLANC	AVG
		English			
Trial One	8.95	24.57	23.78	6.51	15.95
Trial Two	8.61	19.22	17.64	5.57	12.76
Trial Three	8.80	22.13	21.41	6.24	14.65
		Spanish			
Trial One	15.02	24.51	23.12	8.19	17.71
Trial Two	12.12	21.24	20.49	7.33	15.30
Trial Three	14.18	23.47	22.14	7.71	16.88
		Chinese			
Trial One	7.82	16.94	18.13	5.96	12.21
Trial Two	6.20	15.19	16.64	4.67	10.68
Trial Three	7.38	16.68	17.45	5.72	11.81

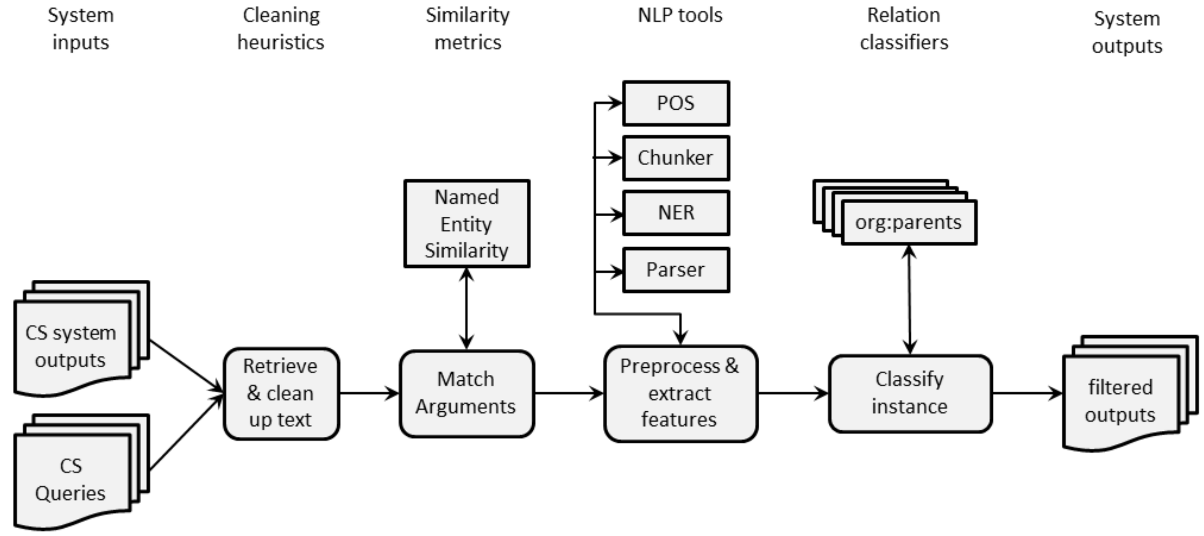

Figure 3: SFV system workflow

to further processing (stripping non-ascii characters, xml/html tags, and text matching a set of filters for ad-hoc formatting). Matches are sought for the filler value in the system output, and the subject of the corresponding query. For each argument, a window of two sentences before and after that matching the relation provenance is scanned, while for the object. This is necessary due to changes in character offsets due to the cleanup step, and in the case of the subject, in accordance with the assessment guidelines. If matches are not found, the query is ignored.

Documents corresponding to surviving queries are processed with a suite of NLP tools to support feature extraction. The feature types follow those described in (Chan and Roth, 2010) and use Part of Speech (Roth and Zelenko, 1998), Shallow Parse (Punyakanok and Roth, 2001), Named Entity (Ratinov and Roth, 2009), and Syntactic Parse (Richard Socher and John Bauer and Christopher D. Manning and Andrew Y. Ng, 2013) representations. After feature extraction, the query is fed to a classifier to determine its label.

To determine whether or not to filter each slot filler output, the Illinois CCG SFV 2016 system uses a set of linear classifiers, one per relation type. Each query is used to generate an example, and the classifier labels the examples as **true** (filter the example) or **false** (retain the example). In previous TAC Slot Filler tasks, systems tend to predict more incorrect relations than correct ones, and SFV systems have generally not improved overall performance because they remove too many correct predictions. We therefore pursue a *cautious* filtering strategy by stratifying the slot fill candidate responses by simple measures of complexity, and by discarding classifiers with low precision (below 85%) on negative examples or which had too few training examples (less than 100). In this way we aim to maintain good overall recall, and improve precision by filtering only examples that we are confident are incorrect.

We experimented with two-stage decision-making where we used predictions from multiple classifiers on the same example as features for a second, meta-classifier; the intuition is that if a second classifier confidently predicts a relation other than that specified by the CSSF query, we should reduce our confidence in the prediction of the classifier for that relation. So far, we have not seen improvements to system performance when we use this method.

3.2 Training Data

We generated training data from the 2015 CSSF assessment data (LDC2015E100), which contains 21,517 hop zero and 9,127 hop one assessments. We treated "inexact" assessments as "correct" for the purposes of this work, as this increases the proportion of positive examples and more closely matches our interpretation of the RTE task.

We stratified the training data based on its perceived simplicity. First, we filtered examples that our SFV system could not reliably transform into a corresponding entailment example (those for which we could not identify corresponding subject or entity in the text within the specified provenance, and those specifying multiple documents as provenance). We split the remaining data into two sets: **conservative** – examples for which we found one exact match for the subject and one exact match for the object in the specified document; and **relaxed** – examples for which we found multiple matches for subject and/or object. We excluded any examples that provide multiple documents as provenance, as the intended meaning of multiple provenances implies that inference is needed to link the information from each to determine the slot filler value. We believe that this may require a different representation than the single document responses.

The resulting conservative and relaxed example sets contain a total of 2,720 and 5,687 examples respectively. These are used to train a set of SVM classifiers, one for each SF relation.

3.3 Experiments

Two sets of per-relation models were trained using SVM: the first used just the **conservative** examples, and the second combined both **conservative** and **relaxed** examples. In each case, we ran five-fold cross-validation to assess model performance. Folds were randomly selected and no effort was made to balance their distributions over positive vs. negative or over relation types to reflect the overall data set. Of the 67 relations (including inverses) represented in the CSSF task, 47 were used by the conservative model, and 22 were used by the relaxed model.

Tables 10 and 11 collect the 5-fold cross-validation results on the examples we extracted from the 2015 CSSF assessment data for the **conservative** and **conservative + relaxed** data sets respectively. For this overview, we report only the average values and for a couple of representative relations. Performance on the majority of relations is strong, though recall that this is on a subset of the slot filler system outputs.

The conservative model was used to process the 2015 SFV data provided in the TAC/LDC release LDC2015E100, which was made available to task participants. Table 12 shows the average performance of the Illinois SFV system on this data, based on the official 2015 assessments (note that this is a subset of the data). Since we are using the assessment data to train the system, this evaluation is mainly useful to verify that the end-to-end system behavior on the Cold Start system outputs is consistent with the model behavior on the examples extracted directly from the gold assessments.

3.4 Official Results from TAC 2016 SFV Task

Tables 13 and 14 present a summary of the Illinois CCG conservative and relaxed SFV systems' performance on the 2016 SFV task. The conservative and relaxed systems correspond to models trained using the conservative and conservative + relaxed example sets described in section 3.2 respectively. The performance for the conservative system is slightly better than that of the relaxed system.

Overall, the CCG SFV system tends to slightly reduce CSSF system performance, though for a few CSSF runs (such as ICTCAS_OKN_KB) the system slightly improves performance.

Experiment	**Precision**	**Recall**	**F1**
gpe:residents_of_country	0.840	0.824	0.822
org:employees_or_members	0.863	0.906	0.883
Overall	0.766	0.807	0.789

Table 10: 5-fold Cross Validation performance on the **conservative** data set for selected relations, and overall. Fold performance is micro-average over relations, and overall performance is micro-average over folds. Example count for relations and overall are the total over all folds.

Experiment	**Precision**	**Recall**	**F1**
gpe:residents_of_country	0.785	0.736	0.748
org:employees_or_members	0.795	0.953	0.853
Overall	0.715	0.758	0.720

Table 11: 5-fold Cross Validation performance on the **relaxed** data set for selected relations, and overall. Fold performance is micro-average over relations, and overall performance is micro-average over folds.

3.5 Discussion

In modeling the Slot Filler Validation as a recognizing textual entailment (RTE) task, we intentionally restrict the information available to the SFV system, and use no information about the system source or from other system predictions.

This work forms the basis of an incremental approach to solving the Slot Filler Validation task. The effort to formulate a conservative filtering strategy is intended to support a systematic exploration of what is required to perform well on examples where we are confident that we can expect (mostly) correct and informative ancillary NLP annotations, and to allow us to design and investigate strategies to work around errors and gaps in those annotations. While it is not a perfect match for the RTE task, which requires strong performance also on *unseen* relations, it is a sufficiently broad set of relations to research strategies that work across multiple target relations.

Presently, it appears that domain shift is a problem: systems in this year's Cold Start task produced significantly fewer predictions, suggesting that they have more conservative behavior than last year, and so the models trained using last years correct and incorrect predictions is not well suited. This is particularly evident for the *relaxed* system, which results in noticeable degradation of performance.

3.6 Future Work

There are a number of experimental parameters that could be explored based on this initial work.

- Combining models trained with conservative and relaxed examples. This requires a refinement of the SFV system to use different model types for different examples.
- Accounting for common lexico-syntactic patterns, following the approach of (Chan and Roth, 2011).
- Parameter sweeps for the thresholds at which we exclude relation classifiers from the filtering task.

We are conducting error analysis to identify gaps in system capabilities at different stages, such as in detecting argument matches in documents, and to understand the potential of the two-level model.

4 Conclusion

The TAC tasks continue to present a challenge to NLP researchers. Independent of the TAC evaluations themselves, the CSSF assessments and system outputs together provide a valuable resource for the NLP community to develop and evaluate Relation Extraction models. The balance between newswire and discussion forum data encourages development of more robust NLP approaches.

References

David Ahn. 2006. The stages of event extraction. In *Workshop on Annotating and Reasoning About Time and Events*, pages 1–8.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jenifer C. Lai, and Robert L. Mercer. 1992.

Label	**Total Gold**	**Total Predicted**	**Correct Prediction**	**Precision**	**Recall**	**F1**
NO	36641	2446	1892	77.35	5.16	9.68
YES	22118	56313	21564	38.29	97.5	54.99
Unfiltered	58759	58759	22118	37.64	37.64	37.64
All	58759	58759	23456	39.92	39.92	39.92

Table 12: Performance on TAC 2015 Cold Start system outputs corresponding to assessed results, using the *conservative* model.

TAC measure	max. change	**Min. Change**	**Avg. Change**
CSSF Micro-Average Hop 0 $\Delta F1$	0.0001	-0.0588	-0.0225
CSSF Micro-Average Hop 1 $\Delta F1$	0.0107	-0.0458	-0.0050
CSSF Micro-Average All $\Delta F1$	0.0001	-0.055	-0.0162
Max. Micro-Average Hop 0 $\Delta F1$	0.0049	-0.0627	-0.0228
Max. Micro-Average Hop 1 $\Delta F1$	0.0086	-0.0529	-0.0057
Max. Micro-Average All $\Delta F1$	0.0021	-0.0599	-0.0165
Mean Macro-Average Hop 0 $\Delta F1$	0.0008	-0.0216	-0.0046
Mean Macro-Average Hop 1 $\Delta F1$	0.0026	-0.0156	-0.0032
Mean Macro-Average All $\Delta F1$	0.0004	-0.0174	-0.0040

Table 13: Summary of filtering results for Illinois CCG SFV *conservative* system in terms of change in F1

Class-Based n-gram Models of Natural Language. *Computational Linguistics*, 18(4):467–479.

Y. Chan and D. Roth. 2010. Exploiting background knowledge for relation extraction. In *Proc. of the International Conference on Computational Linguistics (COLING)*, Beijing, China, 8.

Y. Chan and D. Roth. 2011. Exploiting syntactico-semantic structures for relation extraction. In *Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL)*, Portland, Oregon.

Chen Chen and Vincent Ng. 2012. Joint modeling for chinese event extraction with rich linguistic features. In *COLING*, pages 529–544.

P. Denis and J. Baldridge. 2007. Joint determination of anaphoricity and coreference resolution using integer programming. In *Proceedings of the Annual Meeting of the North American Association of Computational Linguistics (NAACL)*.

Evgeniy Gabrilovich and Shaul Markovitch. 2005. Feature generation for text categorization using world knowledge. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1048–1053.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Haoruo Peng, Yangqiu Song, and Dan Roth. 2016. Event detection and co-reference with minimal supervision. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

V. Punyakanok and D. Roth. 2001. The use of classifiers in sequential inference. In *Proc. of the Conference on Neural Information Processing Systems (NIPS)*, pages 995–1001. MIT Press.

L. Ratinov and D. Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proc. of the Conference on Computational Natural Language Learning (CoNLL)*, 6.

Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and global algorithms for disambiguation to wikipedia. In *Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Richard Socher and John Bauer and Christopher D. Manning and Andrew Y. Ng. 2013. Parsing With Compositional Vector Grammars. In *Association for Computational Linguistics (ACL)*.

D. Roth and D. Zelenko. 1998. Part of speech tagging using a network of linear separators. In *Coling-Acl, The 17th International Conference on Computational Linguistics*, pages 1136–1142.

M. Sammons, H. Peng, Y. Song, S. Upadhyay, C.-T. Tsai, P. Reddy, S. Roy, and D. Roth. 2015a. Illinois ccg tac 2015 event nugget, entity discovery and linking, and slot filler validation systems. In *TAC*.

Mark Sammons, Haoruo Peng, Yangqiu Song, Shyam Upadhyay, Chen-Tse Tsai, Pavankumar Reddy,

TAC measure	Max. change	**Min. Change**	**Avg. Change**
CSSF Micro-Average Hop 0 $\Delta F1$	0	-0.0698	-0.0238
CSSF Micro-Average Hop 1 $\Delta F1$	0	-0.0546	-0.0099
CSSF Micro-Average All $\Delta F1$	0	-0.0638	-0.0183
Max. Micro-Average Hop 0 $\Delta F1$	0.0049	-0.0627	-0.0228
Max. Micro-Average Hop 1 $\Delta F1$	0.0086	-0.0529	-0.0057
Max. Micro-Average All $\Delta F1$	0.0021	-0.0599	-0.0165
Mean Macro-Average Hop 0 $\Delta F1$	0	-0.0698	-0.0238
Mean Macro-Average Hop 1 $\Delta F1$	0	-0.0546	-0.0099
Mean Macro-Average All $\Delta F1$	0	-0.0638	-0.0183

Table 14: Summary of filtering results for Illinois CCG SFV ”relaxed” system in terms of change in F1

Subhro Roy, and Dan Roth. 2015b. Illinois CCG TAC 2015 Event Nugget, Entity Discovery and Linking, and Slot Filler Validation Systems. In *TAC*.

Chen-Tse Tsai and Dan Roth. 2016. Concept grounding to multiple knowledge bases via indirect supervision. *Transactions of the Association for Computational Linguistics*, 2.

Chen-Tse Tsai, Stephen Mayhew, and Dan Roth. 2016. Cross-lingual named entity recognition via wikification. In *Proc. of the Conference on Computational Natural Language Learning (CoNLL)*.