Dynamic Knowledge-Base Alignment for Coreference Resolution

Jiaping Zheng Luke Vilnis Sameer Singh Jinho D. Choi Andrew McCallum
School of Computer Science
University of Massachusetts
Amherst MA 01003
{jzheng,luke,sameer,jdchoi,mccallum}@cs.umass.edu

Abstract

Coreference resolution systems can benefit greatly from inclusion of *global* context, and a number of recent approaches have demonstrated improvements when precomputing an alignment to external knowledge sources. However, since alignment itself is a challenging task and is often noisy, existing systems either align conservatively, resulting in very few links, or combine the attributes of multiple candidates, leading to a conflation of entities. Our approach instead performs joint inference between within-document coreference and entity linking, maintaining ranked lists of candidate entities that are dynamically merged and reranked during inference. Further, we incorporate a large set of surface string variations for each entity by using anchor texts from the web that link to the entity. These forms of global context enables our system to improve classifier-based coreference by 1.09 B^3 F1 points, and improve over the previous state-of-art by 0.41 points, thus introducing a new state-of-art result on the ACE 2004 data.

1 Introduction

Coreference resolution is the task of identifying sets of noun phrase mentions from a document that refer to the same real-world entities. For example, in the following excerpt: *"The Chicago suburb of Arlington Heights is the first stop for* $\langle$*George W. Bush*$\rangle_1$ *today.* $\langle$*The Texas governor*$\rangle_2$ *stops in* $\langle$*Gore's home state*$\rangle_3$ *of* $\langle$*Tennessee*$\rangle_4$ *this afternoon...* ", (m_1, m_2) and (m_3, m_4) define the coreferent pairs. Coreference resolution forms an important component for natural language processing and information extraction pipelines due to its utility in relation extraction, cross-document coreference, text summarization, and question answering. The task of coreference is challenging for automated systems as the local information contained in the document is often not enough to accurately disambiguate mentions, for example, coreferencing (m_1, m_2) requires identifying that George W. Bush (m_1) is the governor of Texas (m_2), and similarly for (m_3, m_4). External knowledge-bases such as FrameNet (Baker et al., 1998), Wikipedia, Yago (Suchanek et al., 2007), and Freebase (Bollacker et al., 2008), can be used to provide *global context*, and there is a strong need for coreference resolution systems to accurately use such sources for disambiguation.

Incorporating external knowledge bases into coreference has been the subject of active recent research. Ponzetto and Strube (2006) and Ratinov and Roth (2012) precompute a fixed alignment of the mentions to the knowledge base entities. The attributes of these entities are used during coreference by incorporating them in the mention features. Since alignment of mentions to the external entities is itself a difficult task, these systems favor high-precision linking. Unfortunately, this results in fewer alignments, and improvements are only shown on mentions that are easier to align and corefer (such as the *non-transcript documents* in Ratinov and Roth (2012)). Alternatively, Rahman and Ng (2011) link each mention to multiple entities in the knowledge base, improving recall at the cost of lower precision; the attributes of all the linked entities are aggregated as features. Although this approach is more robust to noise in the documents, the features of a mention merge the different aspects of the entities, for example a "Michael Jordan" mention will contain features for both the *scientist* and *basketball* personas.

Instead of fixing the alignment of the mentions to the knowledge base, our proposed approach maintains a ranked list of candidate entities for each mention. To expand the set of surface strings that may be used to refer to each entity, the attributes of each candidate contain anchor texts (the visible text) of the links on the web that refer to that entity candidate. When mentions are compared during inference, we use the features computed from the top ranked entity candidate of the antecedent mention. As mentions are merged, the ranked lists of candidate entities are also merged and reranked, often changing the top-ranked entity candidate used in subsequent comparisons. The large set of surface string variations and constant reranking of the entity candidates during inference allows our approach to correct mistakes in alignment and makes external information applicable to a wider variety of mentions.

Our paper provides the following contributions: (1) an approach that jointly reasons about both within-doc entities and their alignment to KB-entities by dynamically adjusting a ranked list of candidate alignments, during coreference, (2) Utilization of a larger set of surface string variations for each entity candidate by using links that appear all over the web (Spitkovsky and Chang, 2012), (3) A combination of these approaches that improves upon a competitive baseline without a knowledge base by 1.09 B^3 F1 points on the ACE 2004 data, and outperforms the state-of-the-art coreference system (Stoyanov and Eisner, 2012) by 0.41 B^3 F1 points, and (4) Accurate predictions on documents that are difficult for coreference, such as the *transcript* documents that were omitted from the evaluation in Ratinov and Roth (2012), and documents that contain a large number of mentions.

2 Baseline Pairwise System

In this section we describe a variant of a commonly-used coreference resolution system that does not utilize external knowledge sources. This widely adopted model casts the problem as a series of binary classifications (Soon et al., 2001; Ng and Cardie, 2002; Ponzetto and Strube, 2006; Bengston and Roth, 2008; Stoyanov et al., 2010). Given a document with its mentions, the system iteratively checks each mention m_j for coreference with preceding mentions using a classifier. A coreference link may be created between m_j and one of these preceding mentions using one of the following strategies. The ClosestLink (Soon et al., 2001) method picks the closest mention to m_j that is positively classified, while the BestLink (Ng and Cardie, 2002) method links m_j to the preceding mention that was scored the highest. If none of the preceding mentions are classified as positive (for ClosestLink), or are above a threshold (for BestLink), then m_j is left unlinked. After all the mentions have been processed, the links are used to generate a transitive closure that corresponds to the recognized entities in the document.

Types	Features
String-Similarity	mention string match, head string match, head substring match, head word pair, *mention substring match*, *acronym*
Syntax	number match, gender match, apposition, relative pronoun, mention type, modifier match, *head word POS tags*
Semantic	synonym, antonym, hypernym, modifier relations, both mentions are surrounded by a verb meaning “to say”, *demonym match*
Other	predicted entity type, predicted entity type match, both mentions in same sentence, *sentence/token distance*, *capitalization*

Table 1: Features of the baseline model. Extensions to Bengston and Roth (2008) are *italicized*.

2.1 Pairwise Mention Features

The features used to train our classifier are similar to those in Bengston and Roth (2008), including lexical, syntactical, semantic, predicted NER types, etc., with the exclusion of their “learned features” that require additional classifiers. Further, we include features that compare the mention strings, the distance between the two mentions in terms of the number of sentences and tokens, and the POS tags of the head words. We also use the conjunctions of these features as in Bengston and Roth (2008), as well as the BestLink approach. The complete set of features are listed in Table 1.

The training for our system is similar to Bengston and Roth (2008). The positive training examples are generated from mentions and their immediate preceding antecedent. The negative examples are generated from mentions and all their preceding non-coreferent mentions. If the mention is not a pronoun, preceding pronouns are not used to create training examples, and they are also excluded during inference. In contrast to averaged perceptron used in Bengston and Roth (2008), our baseline system is trained using hinge-loss, ℓ_2-regularized SVM.

2.2 Merging Pairwise Features

When a mention m_j is compared against a preceding mention m_i, information from other mentions that are already coreferent with m_i may be helpful in disambiguating m_j as they may contain information that is not available from m_i. Let M be the mentions between m_i and m_j that are coreferent with m_i. Let $m_q \in M$ be the mention that is closest to m_j. All the features from the pair (m_q, m_j), except those that characterize one mention (for example, mention type of m_j), are added to the features between (m_i, m_j). This extends a similar approach by Lee et al. (2011) that merges only the attributes of mentions (such as gender, but not all pairwise features).

2.3 Pruning Comparisons During Training

A potential drawback of including all the negative examples as in Bengston and Roth (2008) is that the negative instances far outnumber the positive ones, which is challenging for training a classifier. In their system, the positive training examples only constitute 1.6% of the total training instances. By contrast, Soon et al. (2001) reduce the number of negative instances by using only mentions between the mention and its closest coreferent pair as negative examples. Instead of just using the closest coreferent mention, we extend this approach to use the k closest of coreferent preceding mentions, where k is tuned using the development data.

3 Dynamic Linking to Knowledge-Base

In this section, we describe our approach to coreference resolution that incorporates external knowledge sources. The approach is an extension of the pairwise model described earlier, with the inclusion of a ranked list of entities, and using a larger set of surface string variations.

3.1 Algorithm

We describe our overall approach in Algorithm 1. The system assumes that the data is annotated with true mention boundaries and mention types. We additionally tokenize the document text and tag the tokens with their parts of speech for use as features. First, an empty entity candidate list is created for each mention in the document. For each proper noun mention, we query a knowledge base for an ordered list of Wikipedia articles that may refer to it, and add these to the mention's candidate list. Other mentions' candidates lists are left empty.

Algorithm 1 Dynamic Linking to Wikipedia

```
Input: Mentions $\{m_j\}$
Initialize blank entity lists $\{E_m\}$                ▷ Section 3.2
for $m \in$ Proper Noun Mentions do
    LinkWikipedia($m$, $E_m$)                           ▷ Section 3.2
    PopulateEntityAttrs($E_m$)                          ▷ Section 3.3
end for
for $m_i \in$ Mentions do
    Antecedents $\leftarrow \{m_1 ... m_{i-1}\}$
    for $\hat{m} \in$ Antecedents do
        $t \leftarrow$ TopRankedAttrs($E_{\hat{m}}$)     ▷ Section 3.4
        $s \leftarrow$ Score($\hat{m}$, $m_i$, $t$)      ▷ Section 3.4
        Scores$_{\hat{m}} \leftarrow s$
    end for
    $m^* \leftarrow \arg\max_{\hat{m}}$ Scores$_{\hat{m}}$
    if Scores$_{m^*} >$ threshold then
        MarkCoreferent($m^*$, $m_i$)
        MergeEntityLists($E_{m^*}$, $E_{m_i}$)          ▷ Section 3.4
    end if
end for
return Coreferent mention clusters
```

After this pre-processing, each mention m_i is compared against its preceding mentions $m_1 \dots m_{i-1}$ and their top-ranked entity candidate using a classifier. Amongst antecedents $m_1 \dots m_{i-1}$ that score above a threshold, the highest-scoring one m_j is marked as coreferent with m_i and the two candidate lists that correspond to m_i and m_j are merged. Merging two mentions results in the merging and reranking of their respective entity candidate lists, described below. If no antecedents score above a threshold, we leave the mention in its singleton cluster.

3.2 Linking to Wikipedia

To create the initial entity candidate lists for proper noun mentions, we query a knowledge base searcher (Dalton and Dietz, 2013) with the text of these mentions. These queries return scored, ranked lists of entity candidates (Wikipedia articles), which we associate with each proper noun mention, leaving the rest of the candidate lists empty. Linking is often noisy, so only selecting the high-precision links as in Ratinov and Roth (2012) results in too few matches, while picking an aggregation of all links results in more noise due to lower precision (Rahman and Ng, 2011). Additionally, since linking is often performed in pre-processing, two mentions that are determined coreferent during inference could still be linked to different KB entities. To avoid these problems, we keep a list of candidate links for each mention, merging the lists when two mentions are determined coreferent, and rerank this list during inference.

3.3 Populating Entity Attributes

After linking to Wikipedia, we have a list of candidate KB entities for each mention. Each entity has access to external information keyed on the Wikipedia article, but this information could more generally come from any knowledge base. Given these entities, there are many possible features that may be used for disambiguation of the mentions, such as *gender* and fine-grained Wikipedia categories as used by Ratinov and Roth (2012), however most of these features may not be relevant to the task of within-document coreference. Instead, an important resource for linking non-proper mentions of an entity is to identify the possible name variations of the entity. For example, it would be useful to know that *Massachusetts* is also referred to as "The 6th State", however this information is not readily available from Wikipedia.[1]

We instead use the corpus described in Spitkovsky and Chang (2012) that consists of anchor texts of links to Wikipedia that appear on web pages. This collection of anchor texts is sufficiently extensive to cover many common misspellings of entity names, as well as many name variations missing from Wikipedia. For example, for the entity "Massachusetts", our anchor texts include misspellings like "Massachussetts" and "Messuchusetts", and the (debatably) affectionate nickname of "Taxachusetts"—none of which are found in Wikipedia. Using these anchor texts, each entity candidate provides a rich set of name variations that we use for disambiguation, as described in the next section.

3.4 Inference with Dynamic Linking

The input to our inference algorithm consists of a number of mentions, a list of ranked entity candidates for the proper noun mentions that are present in the KB, and a list of attributes (in this case, name variations) for each entity candidate.

Scoring: Our underlying model is a pairwise classification approach as described in Section 2. Similar to existing coreference systems such as Bengston and Roth (2008) and Rahman and Ng (2011), we perform coreference resolution using greedy left-to-right pairwise mention classification, clustering each mention with its highest-scoring antecedent (or leaving it as a singleton temporarily if no score is above a threshold). We add the same additional features and perform feature merging operation (Section 2.2) as in our baseline system.

The top-ranked entity candidate of the antecedent mention is used during coreference to provide additional features for the pairwise classifier. Only using the top-ranked entity candidate allows the system to maintain a consistent *one entity per cluster* hypothesis, reducing the noise resulting from conflated entities. The attributes for this top-ranked entity consist of name variations. We add a binary feature, and conjunctions of this with other features, if the text of the right mention matches one of these name variations.

Entity List Merging: Once a mention pair is scored as coreferent, their corresponding entity candidates are merged. Merging is performed by simply combining the two lists of candidates. Note that there is only one candidate list for a given group of coreferent mentions at any point in inference: if m_1 and m_2 have been previously marked as coreferent, and m_3 is marked as coreferent with m_2, m_1's entity candidates will then contain those from m_3 for future classification decisions.

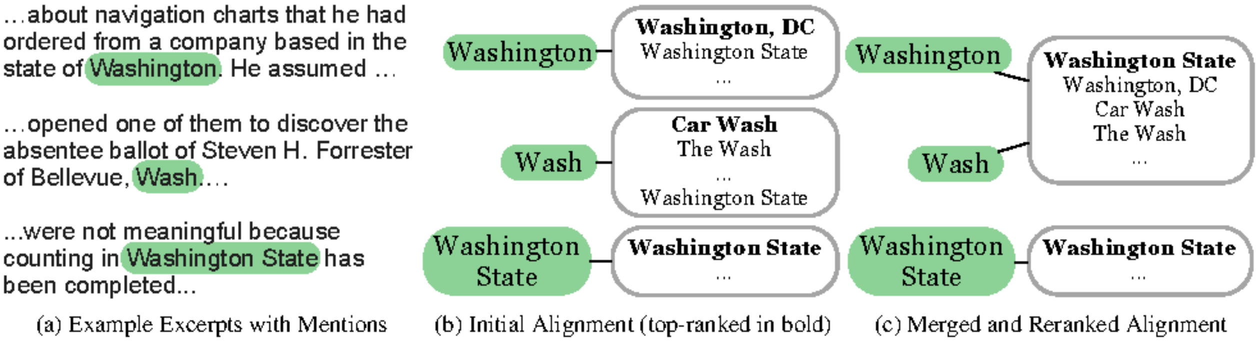
Re-Ranking: After the two entity candidate lists are merged, we rerank the candidates to identify the top-ranked one. We sort the new list of candidate entities by the number of times each candidate occurs in the list, breaking ties by their original relevance from the KB. For example, if two mentions disagree on the top-ranked KB search result, but agree on the second one, after being clustered they will both use the second search result when creating feature vectors for future coreference decisions. Even though other candidates besides the top-ranked one are ignored for a single classification decision, they may become top-ranked after merging with later candidate sets.

This approach allows our system to use the intermediate results of coreference resolution to re-link mentions to KB entities, reducing the noise and contradictory features from incorrect links. Additionally, features from the KB are added to non-proper noun mentions once those mentions are linked with a populated entity, allowing the results of coreference to enrich non-proper noun mentions with KB-based features. The initial proper noun queries effectively seed the linking process, and KB data is then dynamically spread to the other mentions through coreference.

3.5 Example

We describe a run of our approach on an example in Figure 1. Consider three mentions, each

[1] Some of this information is available as *redirects* and from links within Wikipedia, however these do not accurately reflect all the variations of the name.


…about navigation charts that he had ordered from a company based in the state of Washington. He assumed …
…opened one of them to discover the absentee ballot of Steven H. Forrester of Bellevue, Wash.…
...were not meaningful because counting in Washington State has been completed...
Washington
Washington, DC
Washington State
…
Wash
Car Wash
The Wash
…
Washington State
Washington State
Washington State
…
Washington
Wash
Washington State
Washington, DC
Car Wash
The Wash
…
Washington State
Washington State
…

(a) Example Excerpts with Mentions (b) Initial Alignment (top-ranked in bold) (c) Merged and Reranked Alignment

Figure 1: Example of Dynamic Alignment

paired with a top-ranked KB candidate: "Washington", "Wash", and "Washington State". For the first two mentions, clearly the top entity candidate is incorrect; hence approaches that rely on a fixed alignment will perform poorly. In particular, since "Washington State" mention is not compatible with the top-ranked entities of the first two mentions (*Washington, D.C.* and *Car Wash* respectively), approaches that do not modify the ranking during inference may not resolve them. However, the correct candidate *Washington State* does appear in the candidate entities of the first two mentions, albeit with a lower rank. In our approach, clustering the first two mentions causes the shared candidate *Washington State* to move to the top of the list. The coreference system is now able to easily identify that the "Washington State" mention is compatible with the *Washington State* entity formed by the previous two mentions, providing evidence that the final mention should be clustered with either of them in subsequent comparisons.

4 Experiments

4.1 Setup

We evaluate our system on the ACE 2004 annotated dataset (Doddington et al., 2004). Following the setup in Bengston and Roth (2008), we split the corpus into training, development, and test sets, resulting in 268 documents in the train set, 107 documents in the test set, and 68 documents in the development set. The data is processed using standard open source tools to segment the sentences and tokenize the corpus, and using the OpenNLP[2] tagger to obtain the POS tags. The hyperparameters of our system, such as regularization, initial number of candidates, and the number of comparisons during training (k in Section 2.3) are tuned on the development data when trained on the train set. The models we use to evaluate on the test data set are trained on the training and development sets, following the standard evaluation for coreference first used by Culotta et al. (2007).

To provide the initial ranked list of entity candidates from Wikipedia, we query the *KB Bridge* system (Dalton and Dietz, 2013) with the proper name mentions. *KB Bridge* is an information-retrieval-based entity linking system that connects the query mentions to Wikipedia entities using a sequential dependence model. This system has been shown to match or outperform the top performing systems in the 2012 TAC KBP entity linking task.

4.2 Methods

Our experiments investigate a number of baselines that are similar or identical to existing approaches.

Wikipedia Linking: As a simple baseline, we directly evaluate the quality of the alignment for coreference by merging all pairs of proper noun mentions that share at least one common candidate, as per *KB bridge*. Further, the non-pronoun mentions are linked to these proper nouns if the mention string matches any of the entity titles or anchor texts.

Bengston and Roth (2008): A pairwise coreference model containing a rich set of features, as described and evaluated in Bengston and Roth (2008).

Baseline: Our implementation of a pairwise model that is similar to the approach in Bengston and Roth (2008) with the differences described in Section 2. This is our baseline system that performs coreference without the use of external knowledge. Incidentally, it outperforms Bengston and Roth (2008).

Dynamic linking: This is our complete system as

[2] http://opennlp.apache.org/

described in Section 3, in which the list of candidates associated with each mention is reranked and modified during inference.

Static linking: Identical to *dynamic linking* except that entity candidate lists are not merged during inference (i.e., Algorithm 1 without line 17). This approach is comparable to the fixed alignment model, as in the approaches of Ponzetto and Strube (2006) and Ratinov and Roth (2012).

4.3 Results

As in Bengston and Roth (2008), we evaluate our system primarily using the B^3 metric (Bagga and Baldwin, 1998), but also include pairwise, MUC and CEAF(m) metrics. The performance of our systems on the test data set is shown in Table 2. These results use true mentions provided in the dataset, since, as suggested by Ng (2010), coreference resolvers that use different mention detectors (extraction from parse tree, detector trained from gold boundaries, etc.) should not be compared.

Our baseline system outperforms Bengston and Roth (2008) by 0.32 B^3 F1 points on this data set. Incorporating Wikipedia and anchor text information from the web with a fixed alignment (static linking) further improves our performance by 0.54 B^3 F1 points. Using dynamic linking, which improves the alignment during inference, achieves another 0.55 F1 point improvement, which is 1.09 F1 above our baseline, 1.41 F1 above the current best pairwise classification system (corresponding to an error reduction of 7.4%), and 0.4 F1 above the current state-of-art on this dataset (Stoyanov and Eisner, 2012). The improvement of the dynamic linking approach over our baselines is consistent across the various evaluation metrics.

5 Discussion

We also explore our system's performance on subsets of the ACE dataset, and on the OntoNotes dataset.

5.1 Document Length

Coreference becomes more difficult as the number of mentions is increased since the number of pairwise comparisons increases quadratically with the number of mentions. We observe this phenomenon in our dataset: the performance on the smallest third of the documents (when sorted according to number of mentions) is 8.5-10% higher than on the largest third of the documents, as per the B^3 metric.

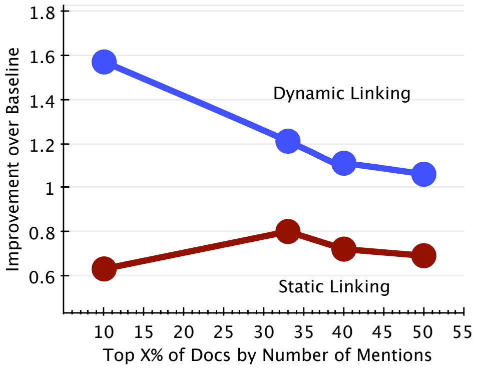


Figure 2: Improvements on the top $X\%$ of documents ranked by the number of mentions.

Method	**Non-Transcripts**	**Transcripts**
Baseline	82.50	79.77
RR 2012	83.03	-
Static Linking	83.06	80.25
Dynamic Linking	83.32	81.13

Table 3: B^3 F1 accuracy on transcripts and non-transcripts from the ACE test data. RR 2012 only evaluate on non-transcripts.

However, we expect dynamic linking of entities to be more beneficial on these larger documents as our system can use the information from a larger number of mentions to improve the alignment during inference. Static linking, on the other hand, is unlikely to obtain higher improvements with the larger number of mentions in the document as the alignment is fixed.

We perform the following experiment to analyze the performance with varying numbers of mentions. We sort all the documents in the test set according to their number of mentions, and evaluate on the top $X\%$ of this list (where X is $10, 33, 40, 50$). As the results demonstrate in Figure 2, the improvement of the static linking approach stays fairly even as X is varied. Even though the experiments suggest that the larger documents are tougher to coreference,[3] dynamic linking provides higher improvements when the documents contain a larger number of mentions.

5.2 Performance on Transcripts

The quality of alignment and the coreference predictions for a document is influenced by the quality of the mentions in the document. In particular,

[3] i.e., the absolute values are lower for these splits. The baseline system obtains 83.08, 79.29, 79.64, and 79.77 respectively for $X = 10, 33, 40, 50$.

Method	Pairwise		MUC		CEAF		B^3	
	P / R	F1	P / R	F1	P / R	F1	P / R	F1
Culotta et al. (2007)	-	-	-	-	-	-	86.7 73.2	79.3
Raghunathan et al. (2010)	71.6 46.2	56.1	80.4 71.8	75.8	-	-	86.3 75.4	80.4
Stoyanov and Eisner (2012)	-	-	-	**80.1**	-	-	-	81.8
Wiki-linking	64.15 14.99	24.30	74.41 28.39	41.10	58.54 58.4	58.47	92.89 57.21	70.81
Bengston and Roth (2008)	-	-	82.7 69.9	75.8	-	-	88.3 74.5	80.8
Baseline	66.56 47.07	55.14	82.84 72.02	77.05	75.58 75.40	75.49	87.02 75.97	81.12
Static Linking	82.53 40.80	54.61	88.39 66.93	76.18	75.33 75.35	75.44	93.10 72.72	81.66
Dynamic Linking	72.20 47.40	**57.23**	85.07 72.02	78.01	76.55 76.37	**76.46**	89.37 76.12	**82.21**

Table 2: Evaluation on the ACE test data, with the system trained on the train and development sets.

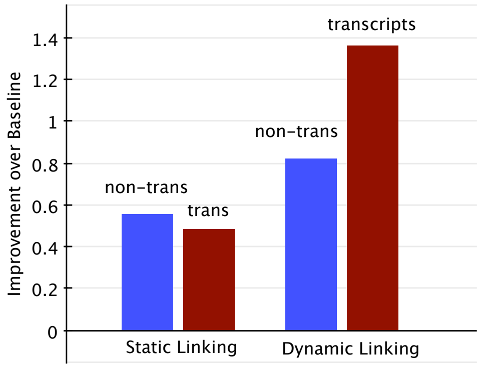


Figure 3: Comparison on the transcripts data.

ACE contains a large number of broadcast news documents, many of which consist of transcribed data containing noise in the form of incomplete sentences and disfluencies. Since these transcripts provide an additional challenge for alignment and coreference, Ratinov and Roth (2012) only use the set of non-transcripts for their evaluation.

Using dynamic linking and a large set of surface string variations, our approach may be able to provide an improvement even on the transcripts. To identify the transcripts in the test set, we use the approximation from Ratinov and Roth (2012) that considers a document to be non-transcribed if it contains proper noun mentions and at least a third of those start with a capital letter. The performance is shown in Table 3, while the improvement over our baseline is shown in Figure 3.

Our static linking matches the performance of Ratinov and Roth (2012) on the non-transcripts. Further, the improvement of static linking on the transcripts over the baseline is lower than that on the non-transcript data, suggesting that noisy mentions and text result in poor quality alignment. Dynamic linking, on the other hand, not only outperforms all other systems, but also shows a higher improvement over the baseline on the transcripts than on non-transcripts. This indicates that dynamic linking approach is robust to noise, and its wider variety of surface strings and flexible alignments are especially useful for transcripts.

5.3 OntoNotes

We also run our systems on the OntoNotes dataset, which was used for evaluation in CoNLL 2011 Shared Task (Pradhan et al., 2011). The dataset consists of 2083 documents from a much larger variety of genres, such as conversations, magazines, web text, etc. Further, the dataset also consists of mentions that refer to events, most of which do not appear as Wikipedia pages. Since only the non-singleton mentions are annotated in the training set, we also include additional noun phrase mentions during training. We obtain B^3 F1 of 65.3, 67.6, and 67.7 for our baseline, static linking, and dynamic linking respectively.[4] When compared to the participants of the closed task, the dynamic linking system outperforms all but two on this metric, suggesting that dynamic alignment is beneficial even when the features have not been engineered for events or for different genres.

6 Related Work

Within-document coreference has been well-studied for a number of years. A variety of approaches incorporate linguistic knowledge as rules iteratively applied to identify the chains, such as Haghighi and Klein (2009), Raghunathan et al. (2010), Stoyanov et al. (2010). Alternatively (and similar to our approach), others represent this knowledge as *features* in a machine learning model. Early applications of such models include Soon et al. (2001), Ng and Cardie (2002) and (Bengston and Roth, 2008). There are also a number of techniques that represent entities explicitly (Culotta et

[4] with MUC 46.1, 49.9 & 50.1, and CEAF(m) 47.9, 49.6 & 49.8, respectively for baseline, static and dynamic linking.

al., 2007; Wick et al., 2009; Haghighi and Klein, 2010; Stoyanov and Eisner, 2012).

This work is an extension of recent approaches that incorporate external knowledge sources to improve within-document coreference. Ponzetto and Strube (2006) identify Wikipedia candidates for each mention as a preprocessing step, and incorporate them as features in a pairwise model. Our method differs in that we draw such features from entity candidates during inference, and also maintain and update a set of candidate entity links instead of selecting only one. Rahman and Ng (2011) introduce similar features from a more extensive set of knowledge sources (such as YAGO and FrameNet) into a cluster-based model whose features change as inference proceeds. However, the features for each cluster come from a combination of all entities aligned to the cluster mentions. We improve upon this approach by maintaining a list of the candidate entities for each mention cluster, modifying this list during the course of inference, and using features from only the top-ranked candidate at any time. Further, they do not provide a comparison on a standard dataset.

Ratinov and Roth (2012) extend the multi-sieve coreference model (Raghunathan et al., 2010) by identifying at most a single candidate for each mention, and incorporating high-precision attributes extracted from Wikipedia. The high-precision mention-candidate pairings are precomputed and fixed; additionally, the features for an entity are based on the predictions of the previous sieves, thus fixed while a sieve is applied. With these restrictions, they show improvements over the state-of-the-art on a subset of ACE mentions that are more easily aligned to Wikipedia, while our approach demonstrates improvements on the complete set of mentions including the tougher to link mentions from the transcripts.

There are a number of approaches that provide an alignment from mentions in a document to Wikipedia. Wikifier (Ratinov et al., 2011) analyzes the context around the mentions and the entities jointly, and was used to align mentions for coreference in Ratinov and Roth (2012). Dalton and Dietz (2013) introduce an approximation to the above approach, but incorporate retrieval-based supervised reranking that provides multiple candidates and scores; this approach performed competitively on previous TAC-KBP entity linking benchmarks (Dietz and Dalton, 2012). Alignment to an external knowledge-base has improved performance for a number of NLP and information extraction tasks, such as named-entity recognition (Cucerzan, 2007; Han and Zhao, 2009), cross-document coreference (Finin et al., 2009; Singh et al., 2010), and relation-extraction (Riedel et al., 2010; Hoffmann et al., 2011).

7 Conclusions

In this paper, we incorporate external knowledge to improve within-document coreference. Instead of fixing the alignment *a priori*, our approach maintains a ranked list of candidate entities for each mention, and merges and reranks the list during inference. Further, we consider a large set of surface string variations for each entity by using anchor texts from the web. These external sources allow our system to achieve a new state-of-the-art on the ACE data. We also demonstrate improvements on documents that are difficult for alignment and coreference, such as transcripts and documents containing a large number of mentions.

A number of possible avenues for future study are apparent. First, our alignment to a knowledge-base can benefit from more document-aware linking to entities, such as the Wikifier (Ratinov et al., 2011). Second, we would like to augment mention features with additional information available from the knowledge base, such as Wikipedia categorization and gender attributes. We also want to investigate a cluster ranking model, as used in (Rahman and Ng, 2011; Stoyanov and Eisner, 2012), to aggregate the features of all the coreferent mentions as inference progresses.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, in part by DARPA under agreement number FA8750-13-2-0020, in part by NSF medium IIS-0803847 and in part by an award from Google. The U.S. Government is authorized to reproduce and distribute reprint for Governmental purposes notwithstanding any copyright annotation thereon. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and necessarily those of the sponsor.

References

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *International Conference on Language Resources and Evaluation (LREC) Workshop on Linguistics Coreference*, pages 563–566.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98, pages 86–90, Stroudsburg, PA, USA. Association for Computational Linguistics.

Eric Bengston and Dan Roth. 2008. Understanding the value of features for coreference resolution. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, SIGMOD '08, pages 1247–1250, New York, NY, USA. ACM.

Silviu Cucerzan. 2007. Large-scale named entity disambiguation based on Wikipedia data. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 708–716.

Aron Culotta, Michael Wick, and Andrew McCallum. 2007. First-order probabilistic models for coreference resolution. In *North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT)*.

Jeffrey Dalton and Laura Dietz. 2013. A neighborhood relevance model for entity linking. In *Open Research Areas in Information Retrieval (OAIR)*.

Laura Dietz and Jeffrey Dalton. 2012. Across-document neighborhood expansion: UMass at TAC KBP 2012 entity linking. In *Text Analysis Conference (TAC)*.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. 2004. The Automatic Content Extraction (ACE) program–tasks, data, and evaluation. In *Proceedings of LREC*, volume 4, pages 837–840. Citeseer.

Tim Finin, Zareen Syed, James Mayfield, Paul McNamee, and Christine Piatko. 2009. Using Wikitology for cross-document entity coreference resolution. In *AAAI Spring Symposium on Learning by Reading and Learning to Read*.

Aria Haghighi and Dan Klein. 2009. Simple coreference resolution with rich syntactic and semantic features. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1152–1161.

Aria Haghighi and Dan Klein. 2010. Coreference resolution in a modular, entity-centered model. In *North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT)*, pages 385–393.

Xianpei Han and Jun Zhao. 2009. Named entity disambiguation by leveraging Wikipedia semantic knowledge. In *Conference on Information and Knowledge Management (CIKM)*, pages 215–224.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 541–550, Portland, Oregon, USA, June. Association for Computational Linguistics.

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 Computational Natural Language Learning (CoNLL)*, pages 28–34. Association for Computational Linguistics.

Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 104–111.

Vincent Ng. 2010. Supervised noun phrase coreference research: the first fifteen years. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1396–1411, Stroudsburg, PA, USA. Association for Computational Linguistics.

Simone Paolo Ponzetto and Michael Strube. 2006. Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT)*, pages 192–199.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. CoNLL-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Conference on Computational Natural Language Learning (CoNLL)*, pages 1–27.

Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A multi-pass sieve for coreference resolution. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 492–501. Association for Computational Linguistics.

Altaf Rahman and Vincent Ng. 2011. Coreference resolution with world knowledge. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 814–824, Portland, Oregon, USA, June.

L. Ratinov and D. Roth. 2012. Learning-based multi-sieve co-reference resolution with knowledge. In *Empirical Methods in Natural Language Processing (EMNLP)*.

L. Ratinov, D. Roth, D. Downey, and M. Anderson. 2011. Local and global algorithms for disambiguation to wikipedia. In *Annual Meeting of the Association for Computational Linguistics (ACL)*.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Collective cross-document relation extraction without labelled data. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Sameer Singh, Michael L. Wick, and Andrew McCallum. 2010. Distantly labeling data for large scale cross-document coreference. *Computing Research Repository (CoRR)*, abs/1005.4298.

Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544, Dec.

Valentin I. Spitkovsky and Angel X. Chang. 2012. A cross-lingual dictionary for english wikipedia concepts. In *International Conference on Language Resources and Evaluation (LREC)*.

Veselin Stoyanov and Jason Eisner. 2012. Easy-first coreference resolution. In *Computational Linguistics (COLING)*.

Veselin Stoyanov, Claire Cardie, Nathan Gilbert, Ellen Riloff, David Buttler, and David Hysom. 2010. Coreference resolution with reconcile. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 156–161, Stroudsburg, PA, USA. Association for Computational Linguistics.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA. ACM.

Michael Wick, Aron Culotta, Khashayar Rohanimanesh, and Andrew McCallum. 2009. An entity-based model for coreference resolution. In *SIAM International Conference on Data Mining (SDM)*.