Cross-Document Coreference Resolution: A Key Technology for Learning by Reading

James Mayfield[1], David Alexander[1], Bonnie Dorr[1], Jason Eisner[1], Tamer Elsayed[1], Tim Finin[1], Clay Fink[1], Marjorie Freedman[1], Nikesh Garera[1], Paul McNamee[1], Saif Mohammad[1], Douglas Oard[1], Christine Piatko[1], Asad Sayeed[1], Zareen Syed[2], Ralph Weischedel[1], Tan Xu[1] and David Yarowsky[1]

[1] Human Language Technology Center of Excellence, 810 Wyman Park Drive, Baltimore MD 21211
[2] University of Maryland, Baltimore County, Baltimore MD 21250

Abstract

Automatic knowledge base population from text is an important technology for a broad range of approaches to learning by reading. Effective automated knowledge base population depends critically upon coreference resolution of entities across sources. Use of a wide range of features, both those that capture evidence for entity merging and those that argue against merging, can significantly improve machine learning-based cross-document coreference resolution. Results from the Global Entity Detection and Recognition task of the NIST Automated Content Extraction (ACE) 2008 evaluation support this conclusion.

Introduction

Learning by reading requires a system to process many different texts, to combine the information gleaned from those texts into a coherent whole, and to subsequently draw inferences from the extracted information. A natural central component of such a system is a knowledge base, which in our definition is a combination of a database, an expressive conceptual schema, a set of background knowledge, and an inference capability. The ability to place knowledge extracted from text into a knowledge base is therefore a critical component of a knowledge-based approach to learning by reading.

Much research has been devoted to extracting information from individual documents in a way that could support such knowledge base population. Evaluations such as MUC and ACE have supported the development of named entity extraction, relation extraction, temporal expression recognition, etc. However, not as much work has been devoted to the integration of multiple processed documents. A critical aspect of multi-document processing is the ability to recognize when two documents are referring to the same concept. Without such a coreference resolution capability, a learning by reading system would be relegated to learning from a large number of unrelated facts.

In this paper, we describe an approach to cross-document coreference resolution of named entities. Our approach is machine learning-based, using training and test collections for which named entities have already been identified and resolved.

Approach

Cross-document coreference resolution is the identification of entity mentions in different documents that refer to the same underlying entity. An entity is anything that might be referred to; however, for our purposes we will concentrate on *named entities*–those that are mentioned by name (e.g., “Barack Obama”). Such entities may also have nominal mentions (e.g., “the country’s president”), pronominal mentions (e.g., “he”), or additional named mentions (e.g., “Barry”).

Our approach to cross-document entity coreference resolution consists of five primary steps:

1. **Intra-document processing.** Numerous approaches to extracting information from individual documents have been described in the literature. Systems exist to extract named entities, relations, time expressions, events, etc., and to perform coreference resolution on them. We do not contribute to these efforts here; we assume that an extraction system is available that can find mentions of the entities of interest in a single document and tie together those that are coreferent.
2. **Entity pairs filtering.** Our approach calculates features on pairs of entities, not on individual entities. Given a large text collection, the number of candidate pairs might be quite large. For example, given a collection of 10,000 documents each containing mentions of ten named entities, about 10^{10} pairs are possible. To reduce the number of pairs that must be fully featurized, we perform a preliminary pairs filtering step that quickly eliminates those pairs that have little chance of being deemed coreferent. For example, with no prior information that they might refer to the same person, an entity with the single name mention ‘George Bush’ and another entity with the single name mention ‘Sojourner Truth’ might be safely ignored.

3. **Featurization**. We calculate a variety of features for each pair of entities. For example, one of the strongest features is the degree to which the mention strings for the two entities match.
4. **Classification**. Using the features calculated for a given pair, the pair is classified as either coreferent or not coreferent. We use machine learning over a set of training examples to perform this classification.
5. **Clustering**. Once the individual pairs are classified, they must be clustered to ensure that all mentions of the same entity are placed in the same equivalence class. This might entail negating some of the individual classification decisions.

In this paper, we concentrate on Steps 2 and 3. Our approach can in theory also apply to nominal and pronominal entity mentions, but our evaluation required that each evaluated entity exhibit at least one named mention.

Types of Features

A focus of our research efforts was on the generation of features over pairs of entities. We divide our features into six broad classes: character-level, document-level, metadata, semantic match, knowledge base instance, and knowledge base ontology features. Note that we did not use syntactic features, primarily because we did not have access to Serif's internal parse trees. The feature space can also be divided into those features that provide evidence for coreference, features that provide evidence against coreference, and features that do both.

Character-level features from exact name string matching can provide strong indications of entity similarity; however they must be robust to possible small errors and difference between entity name strings. These features included *exact match* features such as longest mention exact match, some mention exact match, multiple mention exact match, all mention exact match. This category also includes *partial match* features such as Dice score using character bigrams, Dice score, using longest mention character bigrams, and match between the last word of longest string mentions. Matches over nominals and pronominals, including exact match, multiple exact match, all matching, and Dice score of mention strings, also fits here.

Document-level features provide evidence based on similarities between the larger context of pairs of entities. These include word-context features, such as the Dice score of words in the document, the Dice score of words around mentions, the cosine score of words in the document, and the cosine score of words around mentions. The category also includes context features of other entities including Dice score of entities in document, and Dice score of entities around mentions.

Metadata features reflect facts about the documents containing the two entities as a whole. They include whether the documents were originally spoken or written, whether they are primarily news documents, and whether the two entities come from the same document. This category also includes social context features, such as whether the two entities are in the same social circle.

Semantic match features cover matching two entities based on their attributes or relations. For example, if two entities are known to have the same father, they are more likely to be coreferent than if they are not. Likewise, if one entity is male and the other is female, they are unlikely to be coreferent.

Knowledge base instance features capture entity similarities and differences using instance data in the knowledge base, such as known aliases. This kind of feature relies primarily on the a priori acquisition of relevant instance data, although it is also possible to extract appropriate instances from the text collection being processed.

Knowledge base ontology features include features derived from the ontology used for the knowledge base schema, or from a related hierarchy or taxonomy. For instance, such features might be based on Reuters topics, on thesaurus concepts, or on Wikitology [Syed et al. 2008] features. Such features map the entities being compared onto the ontology or hierarchy, then make their comparison in the space defined by the resource. For example, one might map each entity mention chain onto a set of thesaurus topics, then compare those topics to determine a similarity score.

System

We built a cross-document coreference resolution system based on the approach outlined above. This section provides a number of system details.

Within-Document Processing

All of our within-document entity resolution was conducted using BBN's SERIF system [Boschee 2005]. For each document, SERIF produces a set of named entities, each of which has one or more mentions. Only entities that include at least one named mention are used for the ACE evaluation. For the COE's submissions to the ACE 2008 cross-document coreference resolution task, we considered only person-to-person and organization-to-organization decisions, trusting SERIF's within-document coreference analysis (which was estimated to be 90% accurate at top-level entity type assignment). Sometimes SERIF generated entity mentions that overlap the text span of other mentions. Such nested or overlapping entities are not permitted by ACE guidelines. Believing this to be a relatively rare phenomenon we made an arbitrary choice to always select the leftmost entity. However roughly 1.5% of entity pairs were affected, and we might have done better to prefer named mentions specifically.

Pairs Filtering

We developed several approaches to identfying candidate coreferent pairs, taking the union of their output as our list of pairs to receive further processing.

In our first approach, a pair had to satisfy the following criteria: each member of the pair must have the same entity type, that type must be PER or ORG, and the pair must also satisfy one of the following: (1) they share a word that has a soundex equivalent in the other pair member; (2) the pair had high similarity between sets of character n-grams for their longest name mention; or (3) the pair had high character n-gram similarity using all of their name mentions. N-grams were lower-cased skip bi-grams with skips of length 0, 1, or 2 allowed. A non-zero skip was indicated with a '*' character, so a name like 'Elliott' would generate both 'el' and 'e*l' (using the second l), 'e*i', but not 'e*o'. "High" similarity was defined as a Dice coefficient of greater than 0.3. This process took approximately 7 hours and generated 148 million pairs.

Our second pairs filtering approach used minhash/locality sensitive hashing to generate candidate pairs. This approach has been used successfully for tasks such as document similarity and collaborative filtering [Das 2007]. Entity mentions were processed to produce canonicalized strings (downcased and punctuation-stripped). We generated two sets of pair matches based on n-gram (2-gram) character overlaps and alias match sets of the canonicalized strings. As in Das [2007] minhash will put two canonicalized strings in the same cluster with probability equal to their set overlap similarity (e.g., set overlap of 2-grams or aliases). We concatenated p hash keys (p=5 for ngrams and p=2 for aliases) for q clusters (q=200) for higher recall of pair matches. Our choices for p and q were tuned on a smaller collection, and wound up underproducing pairs on the ACE 2008 collection. Future work should consider more effective parameter tuning of p and q for pairs generation in anticipation of unknown collections, matching entities using sets of strings mentions (vs. matching individual strings), and pairs generation based on set matching of the an appropriate subspace of the entire feature space available to the system.

Our third pairs filtering approach captured known aliases. We derived aliases from Freebase, from BBN's name match lists (any pair appearing on the list was used, without reference to the score for the pair), from a list of stock ticker symbols, and by scraping the TDT and ACE 2008 collections for explicitly stated aliases. Any pair that matched a known alias in any name mention was selected for further processing.

The common traits of these three approaches is that they are fast enough to apply to all candidate pairs, and that they produce high recall. Subsequent expensive featurization and classification then ensures that pair precision is increased.

Featurization

Our general approach to featurization was explained above. In this section, we give more details on a sampling of the features we used.

Document similarity features

A useful feature is the degree of similarity between the documents containing the two entities being featurized. However, computing such document similarity is expensive. We parallelized this task using the MapReduce framework. Similarity scores for all types of vectors were computed via the Ivory system which efficiently computes pairwise similarity of a given large collection of text vectors using the Hadoop MapReduce framework [Apache 2008, Elsayed 2008]. On two MapReduce steps, the vectors are first indexed and then each term generates a set of partial contributions for pairs that contain it. The partial contributions are eventually summed for each pair of vectors. A document frequency cutoff was adopted to drop the least informative terms over the whole set of vectors. For each type of vectors, we chose a suitable threshold based on the training data. The system was run on a Hadoop cluster of 32 nodes and used to compute the similarity matrices.

Usenet features

Email and other communications are written in a social context. In many cases, it is impossible to make accurate coreference decisions without knowing that context. In the ACE collection, Usenet news articles serve as a stand-in for email (they are similar in that they have explicit senders and recipients, and use informal language in much the same way that email does). We provided two similarity features that are based solely on the Usenet documents. The features aimed to cluster personal entities that have at least one email address used in sending or receiving Usenet posts. We adopted a context expansion technique that is generally suited for informal communication data. The technique, detailed in Elsayed et al. [2008a and 2008b] is designed to resolve the identity of personal-name mentions in email collections. By resolving the identity of a mention, we aim to link it to the email address of its true referent. Our expansion technique makes use of four types of context: the email that includes the mention, the thread that includes such email, other emails that are topically relevant to it, and the other emails sent or received by its participants. In each email in such reconstructed context, other less-ambiguous mentions were used to resolve the concerned mention. A ranking algorithm then ranks candidates based on evidence combined from the context.

In post-hoc analysis of this category of features, we found that only fifteen Usenet documents from the ACE collection were annotated by the assessors. The content of thirteen of these came from standard sources, i.e., newswire. Only two of the annotated Usenet articles were actually written by the sender. Of these, one entity refers to the sender, and there is no pair of co-referent mentions to senders. Thus, the ACE 2008 annotated Usenet data was too close to newswire for genre-specific techniques and was too small for reliable analysis of genre-specific features of social context.

Thesaurus concept features

Certain mentions across different documents may be identical in form but may refer to different entities. For example, 'Alexander' may refer to a Macedonian king in some documents and to the inventor of the modern telephone in others. In such cases, the context of these mentions can be used to distinguish the two entities.

We used the 1000 categories in the Macquarie Thesaurus [Macquarie 2006] as coarse-grained senses or concepts. Mohammad and Hirst [2006] describe a method to estimate the strength of co-ccurrence association between a word and a concept from an unannotated corpus (and without the use of a sense-annotated corpus). We used a modification of this approach to determine the strength of co-occurrence association between a concept and the set of words around target mentions. The strength of association between the concepts and the contexts of target mentions is used to represent the target mentions in concept space.

Biographical features

Garera and Yarowsky have developed novel techniques for extracting biographical attributes from text. They perform arbitrary relation extraction using modeling and bootstrapping. This technique can work for arbitrary new attributes and relations of potential interest. No direct guidance is required on the nature or properties of the attributes, beyond seed examples of the desired relationships. The technique works for data in any language with little or no language-specific expertise. The technique models the domain of the attribute space, finds instantiations in large text collections, and models linkages between attributes. By building linkage and context models, estimates can be found for biographical attributes such as P(E "worked as an" A). We used these techniques to assess agreement between entity pairs on the biographic features of sex, nationality, spouse, parent, sibling, occupation, and occupation. In addition to exact match, we used a fuzzy match for occupation (e.g., *lawyer* is similar to *attorney*). Agreement provided either positive evidence for match (e.g., when two mentions have same occupation) or negative evidence for match (e.g., when two entities

Wikitology features

Wikitology [Syed et al, 2008] is a taxonomy derived from the pages of Wikipedia. We used a version of the Wikitology system as a knowledge base of known individuals and organizations as well as general concepts. We defined twelve features based on Wikitology, seven intended to measure similarity and five to measure dissimilarity. Further details on these features may be found in another paper in the proceedings of this Symposium [Finin et al. 2009].

Classification

We explored two types of learning algorithms: support vector machines (using SVM-Perf [Joachims 2005]) and decision trees (using C4.5). In experiments on our test sets the decision trees tended to over-conflate entities; we therefore used the SVM approach for our official ACE submissions. We used a linear kernel. SVM-Perf was quite efficient in learning, examining millions of vectors and extracting fewer than 25 support vectors in under 20 minutes.

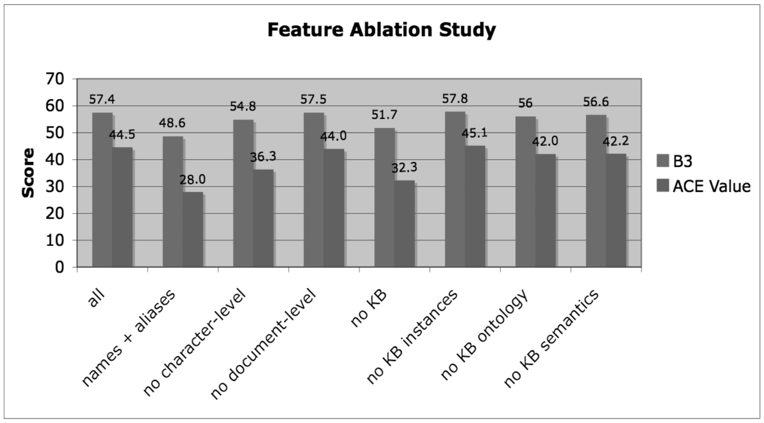
As our core method for the English tasks is based on supervised learning we needed training data on which to construct a classifier to ascertain whether two entity mention chains are coreferent. We used three collections for this purpose:

***A5**: ACE 2005 corpus with MITRE/CLSP annotations.* As part of the JHU 2007 summer workshop on *Exploiting Lexical & Encyclopedic Resources For Entity Disambiguation* [Johns Hopkins 2007] MITRE produced cross-document coreference judgments for named entities appearing in the ACE 2005 data (599 documents of diverse genre). This training corpus was designated "A5." Little name ambiguity is present in this collection; in fact, the simple baseline of grouping together every entity based on exact name match of the longest mention yields a B-Cubed F-score of 0.90. Adding fuzzier name matching, semantic type (i.e., person, organization, geopolitical entity or location), and whether the entities occur in the same file produces a score of 0.96. The MITRE/CLSP annotations contain a number of mistakes. For example Sharon Osbourne and Ariel Sharon are identified as a single entity, and there are two different entities for Colin Powell. Thus perfect performance on this collection is not possible. This data set has the nice property that truth assignments are available for nearly all entities attested in the corpus. The corpus followed the original ACE 2005 partitions into devtrain, devtest, and test.

***A5A**: Ambiguated ACE 2005 corpus.* To make the available data more suitable for the ACE 2008 tasks, we synthetically degraded the ACE 2005 collection in two ways. First, we split person entities with multiple mentions by modifying their name mentions. We applied three kinds of renamings for splitting: nicknames (e.g., renaming *Donald* to *Don*), alternate surname spellings (e.g., renaming *Osbourne* to *Osborn*), and introducing likely misspellings based on QWERTY keyboard placement (e.g., renaming *Neely* to *Heely*). Second, we conflated pairs of distinct entities by giving them the same name in their name mentions, but preserving their separate cluster identifications. We produced a single ambiguated corpus, called "A5A," following the devtrain/devtest/test split of A5.

***WP**: Web People.* The SEMEVAL 2007 workshop on web people disambiguation (WePS) [Artiles 2007] developed a collection of Web pages and of people sharing a name, and judgments on those documents; 79 two-word names (50 training, 29 test) were used. To build the collection, the WePS organizers submitted each name to an Internet search engine, and manually clustered the top 100 result documents according to which person of the target name was mentioned in the document. This is an efficient way to annotate a corpus, requiring only on the order of 79 x 100 human decisions. Two separate annotators were used to ensure accuracy. The collection has ground truth cross-document judgments for the initial set of 79 names, but not for other names occurring in the collection. These data have the advantage that they contain naturally occurring examples of multiple distinct people sharing a name. We converted the available ground truth judgments (which *documents* refer to which people), and assigned a unique cross-document identifier for named entities identified in SERIF analyses of the source text.

Availability of these three collections allowed us to apply machine learning to the English cross-document coreference resolution tasks.



Clustering

We clustered the resulting entity pairs by eliminating any pair with an SVM output weight of less than 0.95, then treating each of the connected components in the resulting graph as a single entity. This approach fares poorly when the classifier mistakenly deems two entities to be coreferent. For example, if the classifier correctly identifies two separate entities in most cases, but makes a single mistake connecting the two, the result will be a single over-conflated entity. A better clustering algorithm is likely to improve the performance of our system significantly.

Evaluation and Results

To evaluate our system, we participated in the ACE 2008 evaluation [NIST 2008a]. While ACE fielded many tasks, we focus here only on the English named entity coreference resolution task. In this task, systems were required to identify named entities in about 11,000 documents of mixed genre, then determine which of these entities are coreferent. Scoring is done using an *ACE Value* metric [NIST 2008a]. Our system achieved an ACE Value of 54.8 on this task, which placed among the better results. ACE discourages publication of system/system comparisons; please see the official results page [ACE 2008b] for further information.

We found the use of both unambiguous and ambiguous training data advantageous. In posthoc experiments the value of the artificially ambiguated data was less clear. We found that training data not specifically designed for the ACE cross-document training task (Web People) was nonetheless useful.

Post-hoc Feature Ablation Study

We studied the contribution of the different sets of features used in our system by ablating features by major categories. We used name and alias matching, derived from the character level match and KB instance features as a strong baseline. This is also the default approach used by many coreference systems. We then evaluating using no character-level features, no document-level features, no knowledge-base features at all (no KB instances, KB ontology or semantic match features), then specifically no KB instance features, no KB ontology features, and no semantic match features.

Our results may be seen in Figure 1. Note that using any subset of the KB feature categories provided similar benefit. This is likely because the features provided similar evidence.

Analysis of Individual Features

In addition to our ablation study, we studied each feature individually to determine its precision, recall and f1 scores. *Precision* is the percentage of entity pairs that the feature properly classifies as coreferent. *Recall* is the percentage of coreferent pairs properly classified by the feature. F1 is the harmonic mean of precision and recall:

$$f1 = 2PR \big/ (P+R)$$

Scoring on features is performed *after* the pairs filtering step, and only pairs that make it through pairs filtering are used in the answer key. Because not every entity identified by the system is a ground truth entity, we need an alignment step to select the best pairing of entities in our results to entities in the ground truth. Once these steps have been carried out, measuring the precision and recall of each feature is straightforward.

Three kinds of feature perform best under the f1 measure:

1. Variants of exact name match tend to score well in both precision and recall. The feature with highest f1 measure (83.1%) reflects the presence of *some* name mention in one entity that has an exact match in the other.
2. Several of the Wikitology-based features did well, such as the cosine similarity of the vectors of top Wikitology article matches (f1=75.1%), and whether the top Wikitology article for the two entities matches (f1=38.1%).
3. Whether an entity contained a mention that was a known alias of a mention found in the other (f1=47.5%).

Features scoring well on precision but not recall are valuable in the few instances they are applicable. Features with precision above 95% include

- A name mentioned by each entity matches exactly one person in Wikipedia.
- The entities have the same parent.
- The entities have the same spouse.
- All name mentions have an exact match across the two entities.
- The longest named mention has an exact match.

Of course, examination of each feature in isolation does not necessarily assign proper value to each feature. It may well be that combinations of features perform better than any of the features individually.

Conclusions

Cross-document coreference resolution is a key technology for knowledge base population, and therefore for learning by reading. We have argued that a machine learning-based approach to cross-document coreference resolution is viable, and that a wide range of features on pairs of entities are useful to such an approach. The ACE 2009 evaluation allowed us to explore the efficacy of over sixty features, both individually and in groups. The results suggest that string matching is perhaps the most important kind of feature to use, but that features based on prior knowledge are also extremely efficacious. The implication for learning by reading is that the representations of learned information, as well as the prior knowledge base to which they are tied, should be actively exploited to reinforce the reading phase.

References

[Andoni 2008] Alexandr Andoni and Pitotr Indyk. 'Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions.' *Communications of the ACM,* 51(1):117-122. January 2008. mags.acm.org/communications/200801/.

[Apache 2008] *Hadoop Web Site.* hadoop.apache.org/.

[Artiles 2007] Javier Artiles, Julio Gonzalo and Satoshi Sekine. *Web People Search Task at SemEval-2007.* nlp.uned.es/weps/.

[Boschee 2005] E. Boschee, R. Weischedel and A. Zamanian, Automatic Information Extraction, *Proceedings of the 2005 International Conference on Intelligence Analysis*, McLean, VA, 2-4 May 2005.

[Das 2007] Google news personalization: scalable online collaborative filtering. AS Das, M Datar, A Garg, S Rajaram. WWW '07: Proceedings of the 16th International World Wide Web Conference, 2007. www2007.org/papers/paper570.pdf.

[Elsayed 2008a] Tamer Elsayed, Douglas W. Oard and Galileo Namata. 'Resolving personal names in email using context expansion.' *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL 2008),* pp. 91-949. 2008.

[Elsayed 2008b] Tamer Elsayed, Jimmy Lin and Douglas Oard, 'Pairwise document similarity in large collections with MapReduce.' *Proceedings of ACL-08*, pp. 265-268. 2008.

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

[Joachims 2005] Thorsten Joachims, 'A Support Vector Method for Multivariate Performance Measures.' *Proceedings of the International Conference on Machine Learning (ICML)*, 2005.

[Johns Hopkins 2007] CLSP Summer Workshop, *Exploiting Lexical & Encyclopedic Resources For Entity Disambiguation*, 2007. www.clsp.jhu.edu/ws2007/groups/elerfed/

[Macquarie 2008] The Macquarie thesaurus. www.macquarieonline.com.au/thesaurus.html.

[NIST 2008a] *Automatic Content Extraction (ACE) Evaluation*. www.nist.gov/speech/tests/ace/

[NIST 2008b] *NIST 2008 Automatic Content Extraction Evaluation (ACE08) Official Results.* www.nist.gov/speech/tests/ace/2008/doc/ace08_eval_official_results_20080929.html.

[Syed 2008] Zareen Syed, Tim Finin, and Anupam Joshi, 'Wikipedia as an ontology for describing documents.' *Proceedings of the Second International Conference on Weblogs and Social Media*, AAAI Press, March 2008.