Who is Who and What is What: Experiments in Cross-Document Co-Reference

Alex Baron
BBN Technologies
10 Moulton Street
Cambridge, MA 02138
abaron@bbn.com

Marjorie Freedman
BBN Technologies
10 Moulton Street
Cambridge, MA 02138
mfreedma@bbn.com

Abstract

This paper describes a language-independent, scalable system for both challenges of cross-document co-reference: name variation and entity disambiguation. We provide system results from the ACE 2008 evaluation in both English and Arabic. Our English system's accuracy is 8.4% relative better than an exact match baseline (and 14.2% relative better over entities mentioned in more than one document). Unlike previous evaluations, ACE 2008 evaluated both name variation and entity disambiguation over naturally occurring named mentions. An information extraction engine finds document entities in text. We describe how our architecture designed for the 10K document ACE task is scalable to an even larger corpus. Our cross-document approach uses the names of entities to find an initial set of document entities that could refer to the same real world entity and then uses an agglomerative clustering algorithm to disambiguate the potentially co-referent document entities. We analyze how different aspects of our system affect performance using ablation studies over the English evaluation set. In addition to evaluating cross-document co-reference performance, we used the results of the cross-document system to improve the accuracy of within-document extraction, and measured the impact in the ACE 2008 within-document evaluation.

1 Introduction

Cross-document entity co-reference is the problem of identifying whether mentions from different documents refer to the same or distinct entities. There are two principal challenges: the same entity can be referred to by more than one name string (e.g. Mahmoud Abbas and Abu Mazen) and the same name string can be shared by more than one entity (e.g. John Smith). Algorithms for solving the cross-document co-reference problem are necessary for systems that build knowledge bases from text, question answering systems, and watch list applications.

There are several challenges in evaluating and developing systems for the cross-document co-reference task. (1) The annotation process required for evaluation and for training is expensive; an annotator must cluster a large number of entities across a large number of documents. The annotator must read the context around each instance of an entity to make reliable judgments. (2) On randomly selected text, a baseline of exact string match will do quite well, making it difficult to evaluate progress. (3) For a machine, there can easily be a scalability challenge since the system must cluster a large number of entities.

Because of the annotation challenges, many previous studies in cross-document co-reference have focused on only the entity disambiguation problem (where one can use string retrieval to collect many documents that contain same name); or have used artificially ambiguated data.

Section 2 describes related work; section 3 introduces ACE, where the work was evaluated; section 4 describes the underlying information extraction engine; sections 5 and 6 address the challenges of coping with name variation and disambiguating entities; sections 7, 8, and 9 present empirical results, improvement of entity extraction

Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 274–283,
Honolulu, October 2008. ©2008 Association for Computational Linguistics

within documents using cross-document coreference, and a difference in performance on person versus organization entities. Section 10 discusses the scalability challenge. Section 11 concludes.

2 Related Work

Person disambiguation given a person name string. Bagga and Baldwin (1998b) produced one of the first works in cross-document co-reference. Their work presented a vector space model for the problem of entity disambiguation, clustering 197 articles that contained the name 'John Smith'.

Participants in the 2007 Sem-Eval Web People Search(WEPS) task clustered 100-document sets based on which person a name string of interest referenced. WEPS document sets were collected by selecting the top 100 web search results to queries about a name string (Artiles, et al., 2007).

Mann and Yarowsky (2003) and Gooi and Allan (2004) used artificially ambiguous data to allow for much larger experiments in clustering documents around a known person of interest.

Clustering different variants of the same name. Lloyd et. al (2006) use a combination of 'morphological similarity' and 'contextual similarity' to cluster name variants that refer to the same entity.

Clustering and disambiguation. The John Hopkins 2007 Summer Workshop produced a cross-document annotated version of the ACE 2005 corpus (18K document entities, 599 documents) consisting of 5 entity types (Day, et. al, 2007). There was little ambiguity or variation in the corpus. Participants demonstrated that disambiguation improvements could be achieved with a Metropolis-Hastings clustering algorithm. The study assumed human markup of document-level entities.

Our work. The work reported in this paper addresses both entity clustering and name variation for both persons and organizations in a corpus of 10K naturally occurring documents selected to be far richer than the ACE 2005 data by NIST and LDC. We investigated a new approach in both English and Arabic, and evaluated on document-level entities detected by information extraction.

3 ACE Evaluation

NIST's ACE evaluation measures system performance on a predetermined set of entities, relations, and events. For the 2008 global entity detection and recognition task (GEDR)[1], system performance was measured on named instances of person and organization entities. The GEDR task was run over both English and Arabic documents. Participants processed over 10K documents for each language. References were produced for about 400 documents per language (NIST, 2008). The evaluation set included documents from several genres over a 10 year time period. Document counts are provided in Table 1. This evaluation differed from previous community cross-document coreference evaluations in that it (a) covered both organizations and people; (b) required processing a relatively large data set; (c) evaluated entity disambiguation and name variation simultaneously; and (d) measured cross-document co-reference over system-detected document-level entities and mentions.

	English	**Arabic**
broadcast conversation	8	38
broadcast news	72	19
meeting	18	---
newswire	237	314
telephone	18	12
usenet	15	15
weblog	47	14

Table 1: Documents per genre in ACE2008 test set

The evaluation set was selected to include interesting cases for cross-document co-reference (e.g cases with spelling variation and entities with shared names). This is necessary because annotation is difficult to produce and naturally sampled data has a high percentage of entities resolvable with string match. The selection techniques were unknown to ACE participants.

4 Extraction System Overview

Our cross-document co-reference system relies on SERIF, a state-of-the-art information extraction (IE) system (Ramshaw, et. al, 2001) for document-level information extraction. The IE system uses statistically trained models to detect and classify mentions, link mentions into entities, and detect and classify relations and events. English and Arabic SERIF share the same general models, although there are differences in the specific features used by the models. Arabic SERIF does not perform event detection. While Arabic SERIF does

[1] NIST's evaluation of cross-document co-reference.

make use of some morphological features, the cross-document co-reference system, which focused specifically on entity names, does not use these features.

Figure 1 and Figure 2 illustrate the architecture and algorithms of the cross-document co-reference system respectively. Our system separately addresses two aspects of the cross-document co-reference problem: name variation (Section 5) and entity disambiguation (Section 6). This leads to a scalable solution as described in Section 10.

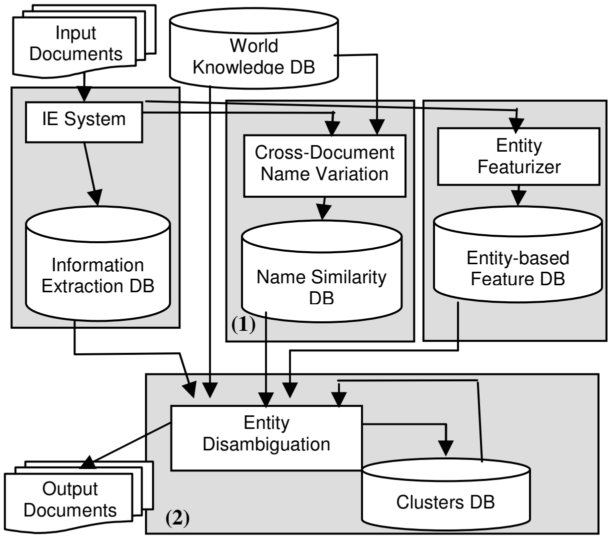

Figure 1: Cross-document Co-reference Architechure

The features used by the cross-document co-reference system can be divided into four classes: World Knowledge (W), String Similarity (S), Predictions about Document Context (C), and Metadata (M). Name variation (V) features operate over unique corpus name strings. Entity disambiguation features (D) operate over document-level entity instances. During disambiguation, the agglomerative clustering algorithm merges two clusters when conditions based on the features are met. For example, two clusters are merged when they share at least half the frequently occurring nouns that describe an entity (e.g. president). As shown in Table 2, features from the same class were often used in both variation and disambiguation. All classes of features were used in both English and Arabic. Because very little training data was available, both the name variation system and the disambiguation system use manually tuned heuristics to combine the features. Tuning was done using the ACE2008 pilot data (LDC, 2008b), documents from the SemEval WEPS task (Artiles, et al., 2007), and some internally annotated documents. Internal annotation was similar in style to the WEPS annotation and did not include full ACE annotation. Annotators simply clustered documents based on potentially confusing entities. Internal annotation was done for ~100 names in both English and Arabic.

Feature Class	Stage	Class
Wikipedia knowledge	D, V	W
Web-mined aliases	V	W
Word-based similarity	D, V	S
Character-based similarity	V	S
Translation dictionaries	V	S
Corpus Mined Aliases	D, V	C
SERIF extraction	D,V	C
Predicted Document Topics	D	C
Metadata (source, date, etc.)	D	M

Table 2: Features for Cross-Document Co-Reference

5 Name Variation

The name variation component (Block 1 of Figure 1) collects all name strings that appear in the document set and provides a measure of similarity between each pair of name strings.[2] Regions (A) and (B) of Figure 2 illustrate the input and output of the name variation component.

This component was initially developed for question answering applications, where when asked the question *'Who is George Bush?'* relevant answers can refer to both George W and George HW (the question is ambiguous). However when asked '*Who leads al Qaeda?'* the QA system must be able to identify spelling variants for the name al Qaeda. For the cross-document co-reference problem, separating the name variation component from the disambiguation component improves the scalability of the system (described in Section 10).

The name variation component makes use of a variety of features including web-mined alias lists, aliases mined from the corpus (e.g *'John aka J'*), statistics about the relations and co-reference decisions predicted by SERIF, character-based edit distance, and token subset trees. The token subset trees algorithm measures similarity using word overlap by building tree-like structures from the unique corpus names based on overlapping tokens. Translation dictionaries (pulled from machine

[2] For the majority of pairs, this similarity score will be 0.

translation training and cross-language links in Wikipedia) account for names that have a canonical form in one language but may appear in many forms in another language.

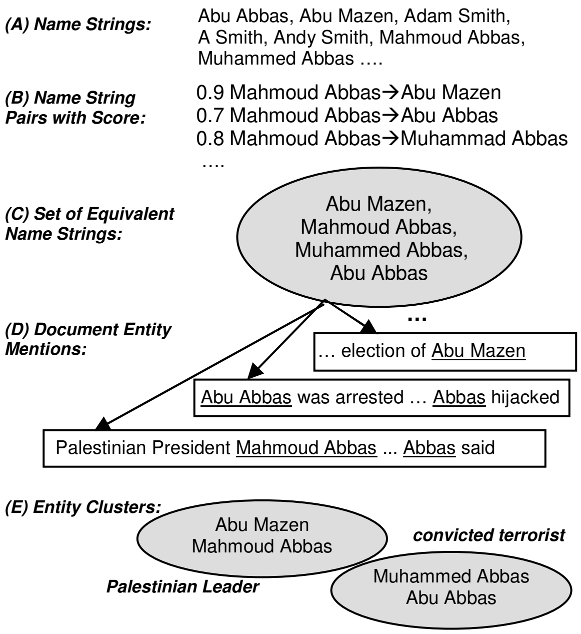
(A) Name Strings:
Abu Abbas, Abu Mazen, Adam Smith, A Smith, Andy Smith, Mahmoud Abbas, Muhammed Abbas ….
(B) Name String Pairs with Score:
0.9 Mahmoud Abbas→Abu Mazen
0.7 Mahmoud Abbas→Abu Abbas
0.8 Mahmoud Abbas→Muhammad Abbas
….
(C) Set of Equivalent Name Strings:
Abu Mazen, Mahmoud Abbas, Muhammed Abbas, Abu Abbas
(D) Document Entity Mentions:
…
… election of Abu Mazen
Abu Abbas was arrested … Abbas hijacked
Palestinian President Mahmoud Abbas ... Abbas said
(E) Entity Clusters:
Abu Mazen Mahmoud Abbas
Palestinian Leader
convicted terrorist
Muhammed Abbas Abu Abbas

Figure 2: Cross-document Co-reference Process

The features are combined with hand-tuned weights resulting in a unidirectional similarity score for each pair of names. The similarity between two name strings is also influenced by the similarity between the contexts in which the two names appear (for example the modifiers or titles that precede a name). This information allows the system to be more lenient with edit distance when the strings appear in a highly similar context, for example increasing the similarity score between *'Iranian President Ahmadinejad'* and *'Iranian President Nejad.'*

6 Entity Disambiguation

We use a complete link agglomerative clustering algorithm for entity disambiguation. To make agglomerative clustering feasible over a 10K document corpus, rather than clustering all document-level entities together, we run agglomerative clustering over subsets of the corpus entities. For each name string, we select the set of names that the variation component chose as valid variants. In Figure 2 region C, we have selected Mahmoud Abbas and 3 variants.

We then run a three stage agglomerative clustering algorithm over the set of document entities that include any of the name string variants or the original name. Figure 2 region D illustrates three document-level entities.

The name variation links are not transitive, and therefore a name string can be associated with more than one clustering instance. Furthermore document-level entities can include more than one name string. However once a document-level entity has been clustered, it remains linked to entities that were a part of that initial clustering. Because of this, the order in which the algorithm selects name strings is important. We sort the name strings so that those names about which we have the most information and believe are less likely to be ambiguous are clustered first. Name strings that are more ambiguous or about which less information is available are clustered later.

The clustering procedure starts by initializing singleton clusters for each document entity, except those document entities that have already participated in an agglomerative clustering process. For those entities that have already been clustered, the clustering algorithm retrieves the existing clusters.

The merging decisions are based on the similarity between two clusters as calculated through feature matches. Many features are designed to capture the context of the document in which entities appear. These features include the document topics (as predicted by the unsupervised topic detection system (Sista, et al., 2002), the publication date and source of a document, and the other names that appear in the document (as predicted by SERIF). Other features are designed to provide information about the specific context in which an entity appears for example: the noun phrases that refer to an entity and the relationships and events in which an entity participates (as predicted by SERIF). Finally some features, such as the uniqueness of a name in Wikipedia are designed to provide the disambiguation component with world knowledge about the entity. Since each cluster represents a global entity, as clusters grow through merges, the features associated with the clusters expand. For example, the set of associated document topics the global entity participates in grows.

While we have experimented with statistically learning the threshold for merging, because of the small amount of available training data, this threshold was set manually for the evaluation.

Clustering over these subsets of similar strings has the additional benefit of limiting the number of global decisions that are affected by a mistake in the within-document entity linking. For example, if in one document, the system linked *Hillary Clinton* to *Bill Clinton*; assuming that the two names are not chosen as similar variants, we are likely to end up with a cluster made largely of mentions of *Hillary* with one spurious mention of *Bill* and a separate cluster that contains all other mentions of *Bill.* In this situation, an agglomerative clustering algorithm that linked over the full set of document-level entities is more likely to be led astray and create a single *'Bill and Hillary'* entity.

7 Experimental Results

Table 3 and Table 4 include preliminary ACE results[3] for the highest, lowest, and average system in the local and cross-document tasks respectively. While a single participant could submit more than one entry, these numbers reflect only the primary submissions. The ACE scorer maps system produced entities to reference entities and produces several metrics. For the within-document task, metrics include ACE Value, B3, and a variant of B3 weighted to reflect ACE value weightings. For the cross-document task, the B3 metric is replaced with F (NIST, 2008). ACE value has traditionally been the official metric of the ACE evaluation. It puts a higher cost on certain classes of entities (e.g. people are more important than facilities), certain classes of mentions (e.g. names are more important than pronouns), and penalizes systems for mistakes in type and subtype detection as well as linking mistakes. Assigning a mention to the wrong entity is very costly in terms of value score. If the mention is a name, a system is penalized 1.0 for the missed mention and an additional 0.75 for a mention false alarm. We will report ACE Value and value weighted B3/F. Scores on the local task are not directly comparable to scores on the global task. The local entity detection and recognition task (LEDR) includes entity detection for five (rather than two) classes of entities and includes pronoun and nominal (e.g. 'the group') mentions in addition to names.

	English		Arabic	
	Val	B3Val	Val	B3Val
Top	52.6	71.5	43.6	69.1
Average	-53.3	50.0	17.3	47.6
Low[4]	-269.1	25.8	-9.1	26.1
BBN-A-edropt	52.1	71.5	43.0	68.9
BBN-B-st-mg	52.6	71.5	43.6	69.1
BBN-B-st-mg-fix[5]	57.2	77.4	44.6	71.3

Table 3: ACE 2008 Within-Document Results (LEDR)

	English		Arabic	
	Val	FVal	Val	FVal
Top	53.0	73.8	28.2	58.7
Average	21.1	59.1	24.7	56.8
Low	-64.1	31.6	21.2	54.8
BBN-B-med	53.0	73.8	28.2	58.7
BBN-B-low	53.2	73.8	28.7	59.3
BBN-B-med-fix[5]	61.7	77	31.4	60.1

Table 4: ACE 2008 Cross-Document Results (GEDR)

Our cross-document co-reference system used BBN-A-edropt as input. BBN-B-st-mg is the result of using cross-document co-reference to improve local results (Section 9). For cross-document co-reference, our primary submission, BBN-B-med, was slightly outperformed by an alternate system BBN-B-low. The two submissions differed only in a parameter setting for the topic detection system (BBN-B-low requires more documents to predict a 'topic'). BBN-A-st-mg-fix and BBN-B-med-fix are the result of post-processing the BBN output to account for a discrepancy between the training and evaluation material.[5]

In addition to releasing results, NIST also released the references. Table 5 includes the ACE score for our submitted English system and the score when the system was run over only the 415 documents with references. The system performs slightly better when operating over the full document set. This suggests that the system is using information from the corpus even when it is not directly scored.

[3] Results in this paper use v2.1 of the references and v17 of the ACE scorer. Final results will be posted to http://www.nist.gov/speech/tests/ace/2008/

[4] There was a swap in rank between metrics, so the low numbers reflect two different systems.

[5] There were discrepancies between the ACE evaluation and training material with respect to the portions of text that should be processed. Therefore our initial system included a number of spurious entities. NIST has accepted revised output that removes these entities. Experiments in this paper reflect the corrected system.

	FVal
10K documents processed (415 scored) (BBN-B-med-fix)	77
Only 415 documents processed	76.3

Table 5: Full English System ACE Evaluation Results

We have run a series of ablation experiments over the 415 files in the English test set to evaluate the effectiveness of different feature classes. These experiments were run using only the annotated files (and not the full 10K document set). We ran two simple baselines. The first baseline (‘No Link’) does not perform any cross-document co-reference, all document entities are independent global entities. The second baseline (‘Exact Match’) links document-level entities using exact string match. We ran 6 variations of our system:

○ Configuration 1 is the most limited system. It uses topics and IE system output for disambiguation, and aliases mined from the documents for the name variation component.

○ Configuration 2 includes Configuration 1 features with the addition of string similarity (edit distance, token subset trees) algorithms for the name variation stage.

○ Configuration 3 includes Configuration 2 features and adds context-based features (e.g. titles and premodifiers) for name variation.

○ Configuration 4 adds information from document metadata to the disambiguation component.

○ Configuration 5 adds web-mined information (alias lists, Wikipedia, etc.) to both the variation and disambiguation components. This is the configuration that was used for our NIST submission.

○ Configuration 5a is identical to Configuration 5 except that the string-based edit distance was removed from the name variation component.

As noted previously, the ACE collection was selected to include challenging entities. The selection criteria of the corpus (which are not known by ACE participants) can affect the importance of features. For example, a corpus that included very few transliterated names would make less use of features based on edit distance.

Figure 3 and Figure 4 show performance (with value weighted F) on the eight conditions over system predicted within-document extraction and reference within-document extraction respectively. Figure 3 also includes configuration 5 run over all 10K documents. We provide two sets of results. The first evaluates system performance over all entities. The relatively high score of the ‘No Link’ baseline indicates that a high percentage of the document-level entities in the corpus are only mentioned in one document. The second set of numbers measures system performance on those entities appearing in more than one reference document. While this metric does not give a complete picture of the cross-document co-reference task (sometimes a singleton entity must be disambiguated from a large entity that shares the same name); it does provide useful insights given the frequency of singleton entities.

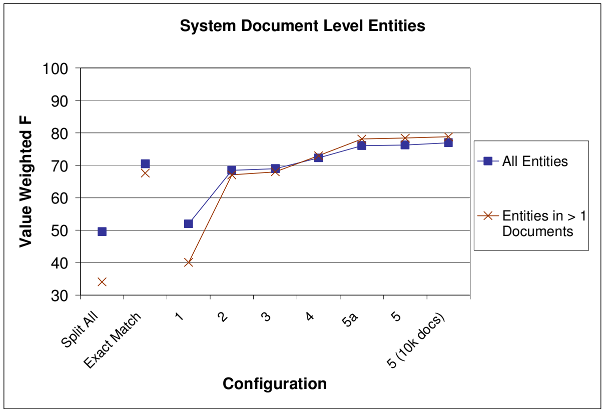

Figure 3: Performance on System Document Entities

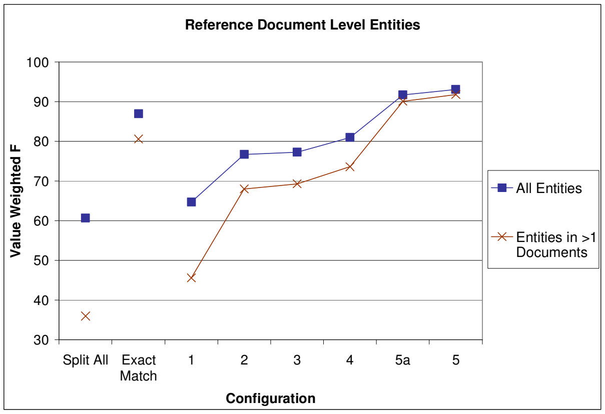

Figure 4: Performance on Perfect Document Entities

Overall system performance improved as features were added. Configuration 1, which disambiguated entities with a small set of features, performed worse than a more aggressive exact string match strategy. The nature of our agglomerative clustering algorithm leads to entity merges only when there is sufficient evidence for the merge. The relatively high performance of the exact match strategy suggests that in the ACE corpus, most entities that shared a name string referred to the same entity, and therefore aggressive merging leads to better performance. As additional features are added, our system becomes more confident and merges more document-level entities.

With the addition of string similarity measures (Configuration 2) our system outperforms the exact match baseline. The submitted results on system entities (Configuration 5) provide a 8.4% relative reduction in error over the exact match baseline. If scored only on entities that occur in more than one document, Configuration 5 gives a 14.2% relative redution in error over the exact match baseline.

The context based features (Configuration 3) allow for more aggressive edit-distance-based name variation when two name strings frequently occur in the same context. In Configuration 3, *'Sheik Hassan Nasrallah'* was a valid variant of *'Hassan Nasrallah'* because both name strings were commonly preceded by '*Hezbollah leader*'. Similarly, *'Dick Cheney'* became a valid variant of '*Richard Bruce Cheney'* because both names were preceded by '*vice president*'. In Configuration 2 the entities included in both sets of name strings had remained unmerged because the strings were not considered valid variants. With the addition of contextual information (Configuration 3), the clustering algorithm created a single global entity. For the '*Dick Cheney'* cluster, this was correct. *'Sheik Hassan Nassrallah'* was a more complex instance, in some cases linking was correct, in others it was not.

The impact of the metadata features (Configuration 4) was both positive and negative. An article about the '*Arab League Secretary General Amru Moussa'* was published on the same day in the same source as an article about '*Intifada Fatah movement leader Abu Moussa*'. With the addition of metadata features, these two distinct global entities were merged. However, the addition of metadata features correctly led to the merging of three instances of the name '*Peter'* in ABC news text (all referring ABC's Peter Jennings).

Web-mined information (Configuration 5) provides several variation and disambiguation features. As we observed, the exact match baseline has fairly high accuracy but is obviously also too aggressive of a strategy. However, for certain very famous global entities, any reference to the name (especially in corpora made of primarily news text) is likely to be a reference to a single global entity. Because these people/organizations are famous, and commonly mentioned, many of the topic and extraction based features will provide insufficient evidence for merging. The same famous person will be mentioned in many different contexts. We use Wikipedia as a resource for such entities. If a name is unambiguous in Wikipedia, then we merge all instances of this name string. In the evaluation corpus, this led to the merging of many different instances of '*Osama Bin Laden'* into a single entity. Web-mined information is also a resource for aliases and acronyms. These alias lists, allowed us to merge '*Abu Muktar'* with '*Khadafi Montanio'* and '*National Liberation Army'* with *'ELN'*.

Interestingly, removing the string edit distance algorithm (System 5a), is a slight improvement over System 5. Initial error analysis has shown that while the string edit distance algorithm did improve accuracy on some entities (e.g linking '*Sam Alito'* with *'Sam Elito'* and linking '*Andres Pastrana'* with '*Andreas Pastrana*'*);* in other cases, the algorithm *allowed* the system to overlink two entities, for example linking *'Megawati Soekarnoputri'* and her sister '*Rachmawati Sukarnoputri'*.

8 Improving Document-Level Extraction with Global Information

In addition to evaluating the cross-document system performance on the GEDR task, we ran a preliminary set of experiments using the cross-document co-reference system to improve within-document extraction. Global output modified within-document extraction in two ways.

First, the cross-document co-reference system was used to modify the within-document system's subtype classification. In addition to evaluating entity links and type classification, the ACE task measures subtype classification. For example, for organization entities, systems distinguish between Media and Entertainment organizations. The IE system uses all mentions in a given entity to assign a subtype. The cross-document co-reference system has merged several document-level entities, and therefore has even more information with which to assign subtypes. The cross-document system also has access to a set of manual labels that have been assigned to Wikipedia categories.

Secondly, we used the cross-document co-reference system's linking decisions to merge within-document entities. If the cross-document co-reference system merged two entities in the same document, then those entities were merged in the within-document output.

Table 6 includes results for our within-document IE system, the IE system with improved subtypes, and the IE system with improved subtypes and merged entities.

	B3Val	Val
Local	77.3	56.7
+ Subtypes	77.3	56.9
+ Merge	77.4	57.2

Table 6: Within-document Results

While these preliminary experiments yield relatively small improvements in accuracy, an analysis of the system's output suggests that the merging approach is quite promising. The output that has been corrected with global merges includes the linking entities with 'World Knowledge' acronyms (e.g. linking *'FARC'* with *'Armed Revolutionary Forces of Colombia');* linking entities despite document-level extraction mistakes (e.g. '*Lady Thatcher*' with '*Margaret Thatcher');* and linking entities despite spelling mistakes in a document (e.g linking *'Avenajado'* with *'Robert Aventajado')*. However, as we have already seen, the cross-document co-reference system does make mistakes and these mistakes can propagate to the within-document output.

In particular, we have noticed that the cross-document system has a tendency to link person names with the same last name when both names appear in a single document. As we think about the set of features used for entity disambiguation, we can see why this would be true. These names may have enough similarity to be considered equivalent names. Because they appear in the same document, they will have the same publication date, document source, and document topics. Adjusting the cross-document system to either use a slightly different approach to cluster document-level entities from the same document or at the very least to be more conservative in applying merges that are the result primarily of document metadata and context to the within-document output could improve accuracy.

9 Effect of LEDR on GEDR

Unlike previous evaluations of cross-document co-reference performance, the ACE 2008 evaluation included both person and organization entities. We have noticed that the performance of the cross-document co-reference system on organizations lags behind the performance of the system on people. In contrast, for LEDR, the extraction system's performance is quite similar between the two entity classes. Furthermore, the difference between global organization and person accuracy in the GEDR is smaller when the GEDR task performed with perfect document-level extraction. Scores are shown in Table 7. These differences suggest that part of the reason for the low performance on organizations in GEDR is within-document accuracy.

	LEDR		GEDR-System		GEDR-Perfect	
	B3Val	Val	FVal	Val	FVal	Val
Org	75.1	51.7	67.8	45.9	91.5	84.0
Per	76.2	52.9	83.2	71.4	94.3	89.5

Table 7: Performance on ORG and PER Entities

The LEDR task evaluates names, nominals, and pronouns. GEDR, however only evaluates over name strings. To see if this was a part of the difference in accuracy, we removed all pronoun and nominal mentions from both the IE system's local output and the reference set. As shown in Table 8, the gap in performance between organizations and people is much larger in this setting.

	LEDR- Name Only	
	B3Val	Val
ORG	82.6	83.0
PER	90.1	90.4

Table 8: Local Performance on Name Only Task

Because the GEDR task focuses exclusively on names and excludes nominals and pronouns, mistakes in mention type labeling (e.g. labeling a name as a nominal) become misses and false alarms rather than type substitutions. As the task is currently defined, type substitutions are much less costly than a missing or false alarm entity.

Intuitively, correctly labeling the name of a person as a name and not a nominal is simple. The distinction for organizations may be fuzzier. For example the string 'the US Department of Justice' could conceivably contain one name, two names, or a name and a nominal. The ACE guidelines (LDC, 2008a) suggest that this distinction can be difficult to make, and in fact have a lengthy set of rules for classifying such cases. However, these rules can seem unintuitive, and may be difficult for machines to learn. For example 'Justice Department' is not a name but 'Department of Justice' is. In some sense, this is an artificial distinction enforced by the task definition, but the accuracy numbers suggest that the distinction has a negative effect on system evaluation.

10 Scalability

One of the challenges for systems participating in the ACE task was the need to process a relatively large document set (10K documents). In question answering applications, our name variation algorithms have been applied to even larger corpora (up to 1M documents). There are two factors that make our solution scalable.

First, much of the name variation work is highly parallelizable. Most of the time spent in this algorithm is spent in the name string edit distance calculation. This is also the only algorithm in the name variation component that scales quadratically with the number of name strings. However, each calculation is independent, and could be done simultaneously (with enough machines). For the 10K document set, we ran this algorithm on one machine, but when working with larger document sets, these computations were run in parallel.

Second, the disambiguation algorithm clusters subsets of document-level entities, rather than running the clustering over all entities in the document set. In the English ACE corpus, the IE system found more than 135K document-level entities that were candidates for global entity resolution. There were 62,516 unique name strings each of which was used to initialize an agglomerative clustering instance. As described in Section 6, a document entity is only clustered one time. Consequently, 36% of these clustering instances are 'skipped' because they contain only already clustered document entities. Even the largest clustering instance contained only 1.4% of the document-level entities.

The vast majority of agglomerative clustering instances disambiguated a small number of document-level entities and ran quickly. 99.7% of the agglomerative clustering runs took less than 1 second. 99.9% took 90 seconds or less.

A small number of clustering instances included a large number of document entities, and took significant time. The largest clustering instance, initialized with the name string '*Xinhua,'* contained 1848 document-level entities (1.4% of the document-level entities in the corpus). This instance took 2.6 hours (27% of the total time spent running agglomerative clustering). Another frequent entity '*George Bush'* took 1.2 hours.

As described in Section 6, the clustering procedure can combine unresolved document-level entities into existing global entities. For large cluster sets (e.g entities referred to by the string '*Xinhua'*), speed would be improved by running many smaller clustering instances on subsets of the document-level entities and then merging the results.

11 Conclusions and Future Work

We have presented a cross-document co-reference clustering algorithm for linking entities across a corpus of documents that

- addresses both the challenges of name variation and entity disambiguation.
- is language-independent,
- is scalable

As measured in ACE 2008, for English our system produced an .8.4% relative reduction in error over a baseline that used exact match of name strings. When measured on only entities that appeared in more than one document, the system gave a 14.2% relative reduction in error. For the Arabic task, our system produced a 7% reduction in error over exact match (12.4% when scored over entities that appear in more than one document). We have shown how a variety of features are important for addressing different aspects of the cross-document co-reference problem. Our current features are merged with hand-tuned weights. As additional development data becomes available, we believe it would be feasible to statistically learn the weights. With statistically learned weights, a larger feature set could improve accuracy even further.

Global information from the cross-document co-reference system improved within-document information extraction. This suggests both that a document-level IE system operating over a large corpus text can improve its accuracy with information that it learns from the corpus; and also that integrating an IE system more closely with a source of world knowledge (e.g. a knowledge base) could improve extraction accuracy.

Acknowledgements

This work was supported by the United States Department of Homeland Security. Elizabeth Boschee and Ralph Weischedel provided useful insights during this work.

References

Artiles, Javier, Julio Gonzalo. & Felisa Verdejo.. 2005. A Testbed for People Searching Strategies. In *the WWW. SIGIR 2005 Conference*. Salvador, Brazil.

Artiles, Javier, Julio Gonzalo. & Satochi Sekine.. 2007. The SemEval-2007 WePS Evaluation: Establishing a benchmark for the Web People Search Task. *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, pages 64–69, Prague, Czech.

Bagga, Amit & Breck Baldwin. 1998a. Algorithms for Scoring Coreference Chains. In *Proceedings of the Linguistic Coreference Workshop at the First International Conference on Language Resources and Evaluation (LREC'98),* pages 563-566.

Bagga, Amit & Breck Baldwin. 1998b. Entity-Based Cross-Document Coreferencing Using the Vector Space Model. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL'98*), pages 79-85.

Day, David.,Jason Duncan, Claudio Guiliano, Rob Hall, Janet Hitzeman,Su Jian, Paul McNamee, Gideon Mann, Stanley Yong & Mike Wick. 2007. CDC Features. *Johns Hopkins Summer Workshop on Cross-Document Entity Disambiguation*. http://www.clsp.jhu.edu/ws2007/groups/elerfed/documents/fullCDED.ppt

Gooi, Chung Heong & James Allan. 2004. Cross-document coreference on a large scale corpus. *In Human Language Technology Conf. North American Chapter Association for Computational Linguistics*, pages 9–16, Boston, Massachusetts, USA.

Lloyd, Levon., Andrew Mehler & Steven Skiena 2006. Identifying Co-referential Names Across Large Corpora. *Combinatorial Pattern Matching*. 2006, pages 12-23, Barcelona, Spain.

Linguistic Data Consortium 2008a. ACE (Automatic Content Extraction) English Annotation Guidelines for Entities Version 6.6 2008.06.13. . Linguistic Data Consortium, Philadelphia. http://projects.ldc.upenn.edu/ace/docs/English-Entities-Guidelines_v6.6.pdf

Linguistic Data Consortium, 2008b. ACE 2008 XDOC Pilot Data V2.1. LDC2007E64. Linguistic Data Consortium, Philadelphia.

Mann, Gideon S. & Yarowsky, David. 2003. Unsupervised Personal Name Disambiguation In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL*, pages 33-40.

NIST Speech Group. 2008. The ACE 2008 evaluation plan: Assessment of Detection and Recognition of Entities and Relations Within and Across Documents. http://www.nist.gov/speech/tests/ace/2008/doc/ace08-evalplan.v1.2d.pdf

Ramshaw, Lance, E. Boschee, S. Bratus, S. Miller, R. Stone, R. Weischedel and A. Zamanian: “Experiments in Multi-Modal Automatic Content Extraction”; in Proc. of HLT-01, San Diego, CA, 2001.

Sista, S, R. Schwartz, T. Leek, and J. Makhoul. An Algorithm for Unsupervised Topic Discovery from Broadcast News Stories. In Proceedings of ACM HLT, San Diego, CA, 2002.