Arabic Cross-Document Person Name Normalization

Walid Magdy, Kareem Darwish, Ossama Emam, and Hany Hassan

Human Language Technologies Group
IBM Cairo Technology Development Center
P.O. Box 166 El-Ahram, Giza, Egypt
{wmagdy, darwishk, emam, hanyh}@eg.ibm.com

Abstract

This paper presents a machine learning approach based on an SVM classifier coupled with preprocessing rules for cross-document named entity normalization. The classifier uses lexical, orthographic, phonetic, and morphological features. The process involves disambiguating different entities with shared name mentions and normalizing identical entities with different name mentions. In evaluating the quality of the clusters, the reported approach achieves a cluster F-measure of 0.93. The approach is significantly better than the two baseline approaches in which none of the entities are normalized or entities with exact name mentions are normalized. The two baseline approaches achieve cluster F-measures of 0.62 and 0.74 respectively. The classifier properly normalizes the vast majority of entities that are misnormalized by the baseline system.

1. Introduction:

Much recent attention has focused on the extraction of salient information from unstructured text. One of the enabling technologies for information extraction is Named Entity Recognition (NER), which is concerned with identifying the names of persons, organizations, locations, expressions of times, quantities, ... etc. (Chinchor, 1999; Maynard et al., 2001; Sekine, 2004; Joachims, 2002). The NER task is challenging due to the ambiguity of natural language and to the lack of uniformity in writing styles and vocabulary used across documents (Solorio, 2004).

Beyond NER, considerable work has focused on the tracking and normalization of entities that could be mentioned using different names (e.g. *George Bush, Bush*) or nominals (e.g. *the president, Mr., the son*) (Florian et al., 2004). Most of the named entity tracking work has focused on intra-document normalization with very limited work on cross-documents normalization.

Recognizing and tracking entities of type "Person Name" are particularly important for information extraction. Yet they pose interesting challenges that require special attention. The problems can result from:

1. A Person's name having many variant spellings (especially when it is transliterated into a foreign language). These variations are typically limited in the same document, but are very common across different documents from different sources (e.g. *Mahmoud Abbas = Mahmod Abas, Mohamed El-Baradei = Muhammad AlBaradey … etc)*.
2. A person having more than one name (e.g. *Mahmoud Abbas = Abu Mazen*).
3. Some names having very similar or identical names but refer to completely different persons (*George H. W. Bush ≠ George W. Bush*).
4. Single token names (e.g. *Bill Clinton = Clinton ≠ Hillary Clinton*).

This paper will focus on Arabic cross-document normalization of named entities of type "person name," which would involve resolving the aforementioned problems. As illustrated in Figure 1, the task involves normalizing a set of person entities into a set of classes each of which is

Proceedings of the 5th Workshop on Important Unresolved Matters, pages 25–32,
Prague, Czech Republic, June 2007.

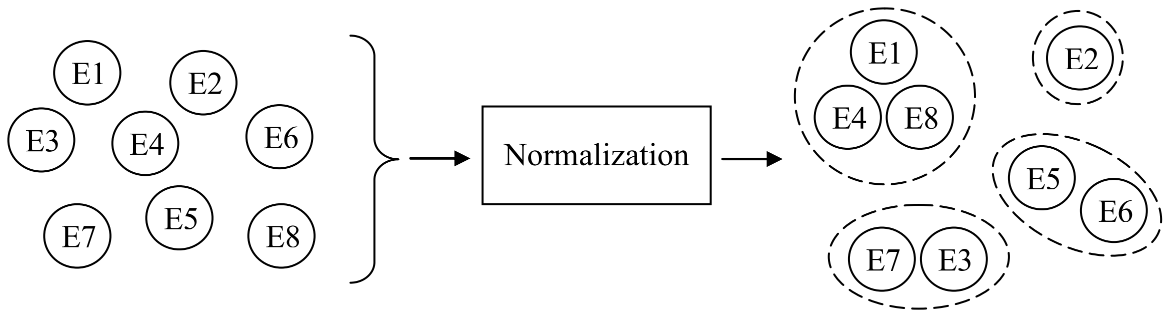

Figure 1 Normalization Model

formed of at least one entity. For N input entities, the output of normalization process will be M classes, where $M \leq N$. Each class would refer to only one person and each class would contain all entities referring to that person.

For this work, intra-document normalization is assumed and an entity refers to a normalized set of name mentions and nominals referring to a single person in a single document. Florian et al. (2004) were kind enough to provide the authors access to an updated version of their state-of-the-art Named Entity Recognition and Tracking (NERT) system, which achieves an F-measure of 0.77 for NER, and an F-measure of 0.88 for intra-document normalization assuming perfect NER. Although the NERT systems is efficient for relatively short documents, it is computational impractical for large documents, which precludes using the NERT system for cross-document normalization through combining the documents into one large document. The main challenges of this work stem from large variations in the spelling of transliterated foreign names and the presence of many common Arabic names (such as Muhammad, Abdullah, Ahmed …etc.), which increases the ambiguity in identifying the person referred to by the mentioned name. Further, the NERT system output system contains many NER errors and intra-document normalization errors.

In this paper, cross-document normalization system employs a two-step approach. In the first step, preprocessing rules are used to remove errant named entities. In the second step, a support vector machine (SVM) classifier is used to determine if two entities from two different documents need to be normalized. The classifier is trained on lexical, orthographic, phonetic, and morphological features.

The paper is organized as follows: Section 2 provides a background on cross-document NE normalization; Section 3 describes the preprocessing steps and data used for training and testing; Section 4 describes the normalization methodology; Section 5 describes the experimental setup; Section 6 reports and discusses experimental results; and Section 7 concludes the paper and provides possible future directions.

2. Background

While considerable work has focused on named entity normalization within a single document, little work has focused on the challenges associated with resolving person name references across multiple documents. Most of the work done in cross-document normalization focused on the problem of determining if two instances with the same name from different documents referring to the same person (Fleischman and Hovy, 2004). Fleischman and Hovy (2004) focused on distinguishing between individuals having identical names, but they did not extend normalization to different names referring to the same individual. Their task is a subtask of what is examined in this paper. They used a large number of features to accomplish their work, depending mostly on language specific dictionaries and wordnet. Some these resources are not available for Arabic and many other languages. Mann and Yarowsky (Mann and Yarowsky, 2003) examined the same problem but they treated it as a clustering task. They focused on information extraction to build biographical profiles (date of birth, place of birth, etc.), and they wanted to disambiguate biographies belonging to different authors with identical names.

Dozier and Zielund (Dozier and Zielund, 2004) reported on cross-document person name normalization in the legal domain. They used a finite state machine that identifies paragraphs in a document containing the names of attorneys, judges, or experts and a semantic parser that extracts from the paragraphs template information about each named individual. They relied on reliable biographies for each individual. A biography would typically contain a person's first name, middle name, last name, firm, city, state, court, and other information. They used a Bayesian network to match the name mentions to the biographical records.

Bhattacharya and Getoor (Bhattacharya and Getoor, 2006) introduced a collective decision algorithm for author name entity resolution, where decisions are not considered on an independent pairwise basis. They focused on using relational links among the references and co-author relationships to infer collaboration groups, which would disambiguate entity names. Such explicit links between co-authors can be extracted directly. However, implicit links can be useful when looking at completely unstructured text. Other work has extended beyond entities of type "person name" to include the normalization of location names (Li et al., 2002) and organizations (Ji and Grishman. 2004).

3. Preprocessing and the Data Set

For this work, a set of 7,184 person name entities was constructed. Building new training and test sets is warranted, because the task at hand is sufficiently different from previously reported tasks in the literature. The entities were recognized from 2,931 topically related documents (relating to the situation in the Gaza and Lebanon during July of 2006) from different Arabic news sources (obtained from searching the Arabic version of news.google.com). The entities were recognized and normalized (within document) using the NERT system of Florian et al (2004). As shown in Figure 2, each entity is composed of a set of name mentions (one or more) and a set of nominal mentions (zero or more).

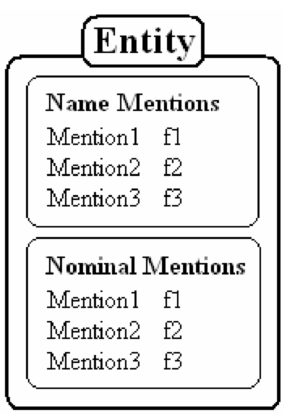

Figure 2 Entity Description

The NERT system achieves an F-score of 0.77 with precision of 0.82 and recall of 0.73 for person name mention and nominal recognition and an F-score of 0.88 for tracking (assuming 100% recognition accuracy). The produced entities may suffer from the following:

1. Errant name mentions: Two name mentions referring to two different entities are concatenated into an errant name mention (e.g. "*Bush Blair*", "*Ahmadinejad Bush*"). These types of errors stem from phrases such as "*The meeting of Bush Blair*" and generally due to lack of sufficient punctuation marks.
2. NE misrecognitions: Regular words are recognized as person name mentions and are embedded into person entities (e.g. *Bush = George Bush = said*).
3. Errant entity tracking: name mentions of different entities are recognized as different mentions of the same entity (e.g. *Bush = Clinton = Ahmadinejad*).
4. Lack of nominal mentions: Many entities do not contain any nominal mentions, which increases the entity ambiguity (especially when there is only one name mention composed of a single token).

To overcome these problems, entities were preprocessed as follows:

1. Errant name mentions such as "*Bush Blair*" were automatically removed. In this step, a dictionary of person name mentions was built from the 2,931 documents collection from which the entities were recognized and normalized along with the frequency of appearance in the collection. For each entity, all its name mentions are checked in the dictionary and their frequencies are compared to each other. Any name mention with a frequency less than 1/30 of the frequency of the name mention with the highest frequency is automatically removed (1/30 was picked based on manual examination of the training set).

2. Name mentions formed of a single token consisting of less than 3 characters are removed. Such names are almost always misrecognized name entities.
3. Name entities with 10 or more different name mentions are automatically removed. The NERT system often produces entities that include many different name mentions referring to different persons as one. Such entities are errant because they over normalize name mentions. Persons are referred to using a limited number of name mentions.
4. Nominal mentions are stemmed using a context sensitive Arabic stemmer (Lee et al. 2003) to overcome the morphological complexity of Arabic. For example, "رئيس" = "president", "الرئيس" = "the president", "والرئيس" = "and the president", "رئيسها" = "its presidents" … etc are stemmed to "رئيس" = "president".

Cross-document entities are compared in a pairwise manner and binary decision is taken on whether they are the same. Therefore, the available 7,184 entities lead to nearly 26 million pairwise comparisons (For N entities, the number of pair wise comparisons $= \frac{N(N-1)}{2}$).

Entity pairs were chosen to be included in the training set if they match any of the following criteria:

1. Both entities have one shared name mention.
2. Both entities have shared nominal mentions.
3. A name mention in one of the entities is a substring of a name mention in the other entity.
4. Both entities have nearly identical name mentions (small edit distance between both mentions).

The resulting set was composed of 19,825 pairs, which were manually judged to determine if they should be normalized or not. These criteria skew the selection of pairs towards more ambiguous cases, which would be better candidates to train the intended SVM classifier, where the items near the boundary dividing the hyperplane are the most important. For the training set, 18,503 pairs were normalized, and 1,322 pairs were judged as different. Unfortunately, the training set selection criteria skewed the distribution of training examples heavily in favor of positive examples. It would interesting to examine other training sets where the distribution of positives and negatives is balanced or skewed in favor of negatives.

The test set was composed of 470 entities that were manually normalized into 253 classes, of which 304 entities were normalized to 87 classes and 166 entities remained unnormalized (forming single-entity classes). Using 470 entities leads to 110,215 pairwise comparisons. The test set, which was distinct from the training set, was chosen using the same criteria as the training set. Further, all duplicate (identical) entities were removed from the test set. The selection criteria insure that the test set is skewed more towards ambiguous cases. Randomly choosing entities would have made the normalization too easy.

4. Normalization Methodology

SVMLight, an SVM classifier (Joachims, 2002), was used for classification with a linear kernel and default parameters. The following training features were employed:

1. The percentage of shared name mentions between two entities calculated as:
   Name Commonality =
   $$\sum_{<common\ names>} \min\left(\frac{f_{1i}}{\sum f_{1j}}, \frac{f_{2i}}{\sum f_{2j}}\right)$$
   where f_{1i} is the frequency of the shared name mention in first entity, and f_{2i} is the frequency of the shared name mention in the second entity. $\sum f_{1i}$ is the number of name mentions appearing in the entity.
2. The maximum number of tokens in the shared name mentions, i.e. if there exists more than one shared name mention then this feature is the number of tokens in the longest shared name mention.
3. The percentage of shared nominal mentions between two entities, and it is calculated as the name commonality but for nominal mentions.
4. The smallest minimum edit distance (Levenshtein distance with uniform weights) between any two name mentions in both entities (Cohen et al., 2003) and this feature is only enabled when name commonality between both entities equals to zero.

5. Phonetic edit distance, which is similar to edit distance except that phonetically similar characters, namely {(ت – t, ط – T), (ك – k, ق – q),(د – d, ض – D),(ث – v, س – s, ص – S), (ذ – *, ز – z, ظ – Z),(ج – j, غ – g),(ـة – p, ـه – h),(إ – <, آ – |, أ – >, ا – A)[1]}, are normalized, vowels are removed, and spaces between tokens are removed.
6. The number of tokens in the pair of name mentions that lead to the minimum edit distance.

Some of the features might seem duplicative. However, the edit distance and phonetic edit distance are often necessary when names are transliterated into Arabic and hence may have different spellings and consequently no shared name mentions. Conversely, given a shared name mention between a pair of entities will lead to zero edit distance, but the name commonality may also be very low indicating two different persons may have a shared name mention. For example "Abdullah the second" and "Abdullah bin Hussein" have the shared name mention "Abdullah" that leads to zero edit distance, but they are in fact two different persons. In this case, the name commonality feature can be indicative of the difference. Further, nominals are important in differentiating between identical name mentions that in fact refer to different persons (Fleischman and Hovy, 2004). The number of tokens feature indicates the importance of the presence of similarity between two name mentions, as the similarity between name mentions formed of one token cannot be indicative for similarity when the number of tokens is more than one.

Further, it is assumed that entities are transitive and are not available all at once, but rather the system has to normalize entities incrementally as they appear. Therefore, for a given set of entity pairs, if the classifier deems that $\text{Entity}_i = \text{Entity}_j$ and $\text{Entity}_j = \text{Entity}_k$, then Entity_i is set to equal Entity_k even if the classifier indicates that $\text{Entity}_i \neq \text{Entity}_k$, and all entities (i, j, and k) are merged into one class.

[1] Buckwalter transliteration scheme is used throughout the paper

5. Experimental Setup

Two baselines were established for the normalization process. In the first, no entities are normalized, which produces single entity classes ("no normalization" condition). In the second, any two entities having two identical name mentions in common are normalized ("surface normalization" condition). For the rest of the experiments, focus was given to two main issues:

1. Determining the effect of the different features used for classification.
2. Determining the effect of varying the number of training examples.

To determine the effect of different features, multiple classifiers were trained using different features, namely:

- All features: all the features mentioned above are used,
- Edit distance removed: edit distance features (features 4, 5, and 6) are removed,
- Number of tokens per name mention removed: the number of shared tokens and the number of tokens leading to the least edit distance (features 2 and 6) are removed.

To determine the effect of training examples, the classifier was trained using all features but with a varying number of training example pairs, namely all 19,825 pairs, a set of randomly picked 5,000 pairs, and a set of randomly picked 2,000 pairs.

For evaluation, 470 entities in test set were normalized into set of classes with different thresholds for the SVM classifier. The quality of the clusters was evaluated using purity, entropy, and Cluster F-measure (CF-measure) in the manner suggested by Rosell et al. (2004). For the cluster quality measures, given cluster i (formed using automatic normalization) and each cluster j (reference normalization formed manually), cluster precision (p) and recall (r) are computed as follows:

$p_{ij} = \frac{n_{ij}}{n_i}$, and $r_{ij} = \frac{n_{ij}}{n_j}$, where n_i number of entities in cluster i, n_j number of entities in cluster j, and n_{ij} number of shared entities between cluster i and j.

The CF-measure for an automatic cluster i against a manually formed reference cluster j is:

$CF_{ij} = \frac{2 \cdot r_{ij} \cdot p_{ij}}{r_{ij} + p_{ij}}$, and the CF-measure for a reference cluster j is:

$CF_j = \max_i \{CF_{ij}\}$.

The final CF-measure is computed over all the reference clusters as follows: $CF = \sum_j \frac{n_{ij}}{n} CF_j$.

Purity of (ρ_i) of an automatically produced cluster i is the maximum cluster precision obtained when comparing it with all the reference clusters as follows: $\rho_i = \max_j \{p_{ij}\}$, and the weighted average purity over all clusters is:

$\rho = \sum_i \frac{n_{ij}}{n_i} \rho_i$, where n is the total number of entities in the set to be normalized (470 in this case).

As for entropy of a cluster, it is calculated as: $E_i = -\sum_j p_{ij} \log p_{ij}$, and the average entropy as:

$E = \sum_i \frac{n_i}{n_i} E_i$.

The CF-measure captures both precision and recall while purity and entropy are precision oriented measures (Rosell et al., 2004).

6. Results and Discussion

Figure 3 shows the purity and CF-measure for the two baseline conditions (no normalization, and surface normalization) and for the normalization system with different SVM thresholds. Since purity is a precision measure, purity is 100% when no normalization is done. The CF-measure is 62% and 74% for baseline runs with no normalization and surface normalization respectively. As can be seen from the results, the baseline run based on exact matching of name mentions in entities achieves low CF-measure and low purity. Low CF-measure values stem from the inability to match identical entities with different name mentions, and the low purity value stems from not disambiguating different entities with shared name mentions. Some notable examples where the surface normalization baseline failed include:

1. The normalization of the different entities referring to the Israeli soldier who is imprisoned in Gaza with different Arabic spellings for his name, namely "جلعاد شليط" (jlEAd $lyT), "جلعاد شاليط" (jlEAd $AlyT), "الجندي شليت" (the soldier $lyt), and so forth.
2. The separation between "الملك عبد الله الثاني" (King Abdullah the Second) and "الملك عبد الله بن عبد العزيز" (King Abdullah ibn Abdul-Aziz) that have a shared name mention "الملك عبدالله" (King Abdullah).
3. The normalization of the different entities representing the president of Palestinian Authority with different name mentions, namely "أبو مازن" (Abu Mazen) and "محمود عباس" (Mahmoud Abbas).

The proposed normalization technique properly normalized the aforementioned examples. Given different SVM thresholds, Figure 3 shows that the purity of resultant classes increases as the SVM threshold increases since the number of normalized entities decreases as the threshold increases. The best CF-measure of 93.1% is obtained at a threshold of 1.4 and as show in Table 1 the corresponding purity and entropy are 97.2% and 0.056 respectively. The results confirm the success of the approach.

Table 1 highlights the effect of removing different training feature and the highest CF-measures (at different SVM thresholds) as a result. The table shows that using all 6 features produced the best results and the removal of the shared names and tokens (features 2 and 6) had the most adverse effect on normalization effectiveness. The adverse effect is reasonable especially given that some single token names such as "*Muhammad*" and "*Abdullah*" are very common and matching one of these names across entities is an insufficient indicator that they are the same. Meanwhile, the exclusion of edit distance features (features 4, 5, and 6) had a lesser but significant adverse impact on normalization effectiveness. Table 1 reports the best results obtained using different thresholds. Perhaps, a separate development set should be used for ascertaining the best threshold.

Table 2 shows that decreasing the number of training examples (all six features are used) has a noticeable but less pronounced effect on normalization effectiveness compared to removing training features.

Table 1 Quality of clusters as measured by purity (higher values are better), entropy (lower values are better), and CF-measure (higher values are better) for different feature sets. Values are shown for max CF-measure. Thresholds were tuned for max CF-measure for each feature configuration separately

Training Data	Purity	Maximum CF-Measure	Entropy	Threshold
No Normalization	100.0%	62.6%	0.000	-
Baseline	83.4%	74.7%	0.151	-
All Features	97.2%	93.1%	0.056	1.4
Edit Distance removed	99.4%	85.5%	0.010	1.0
# of tokens/name removed	96.6%	77.8%	0.071	1.5

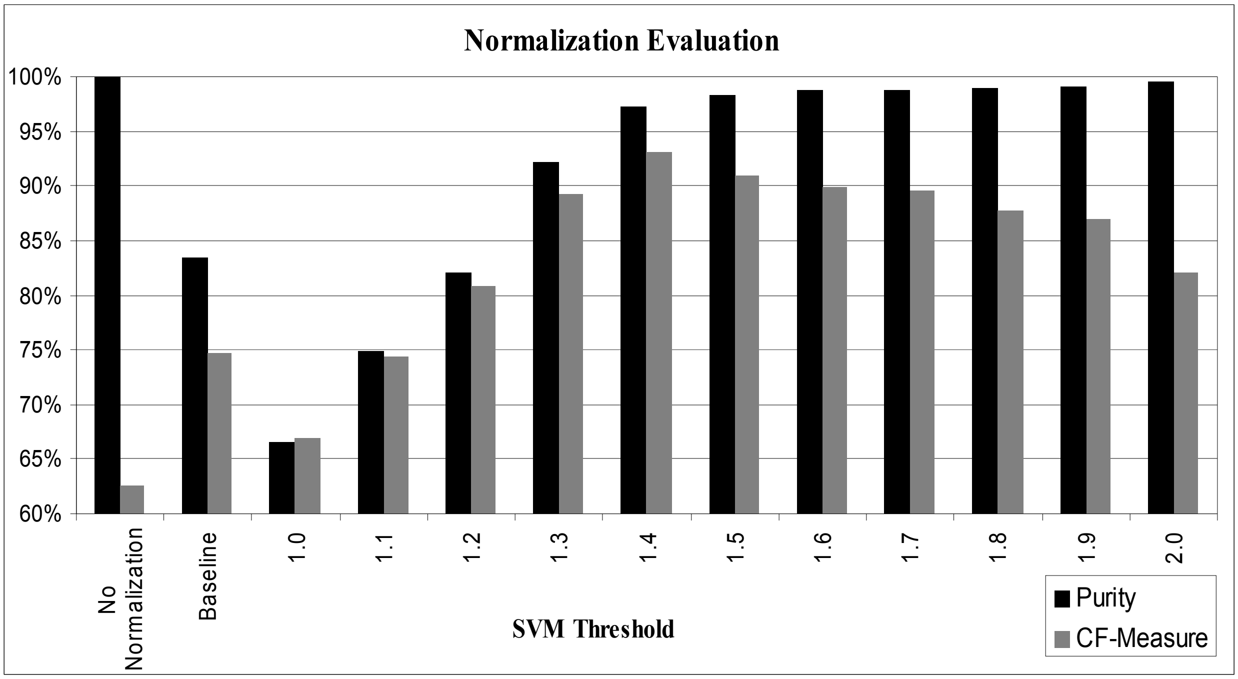

Figure 3 Purity and cluster F-measure versus SVM Threshold

Table 2 Effect of number of training examples on normalization effectiveness

Training Data	Purity	Maximum CF-Measure	Entropy	Threshold
20k training pairs	97.2%	93.1%	0.056	1.4
5k training pairs	97.4%	90.5%	0.053	1.5
2k training pairs	98.5%	90.3%	0.031	1.6

7. Conclusion:

This paper presented a two-step approach to cross-document named entity normalization. In the first step, preprocessing rules are used to remove errant named entities. In the second step, a machine learning approach based on an SVM classifier to disambiguate different entities with matching name mentions and to normalize identical entities with different name mentions. The classifier was trained on features that capture name mentions and nominals overlap between entities, edit distance, and phonetic similarity. In evaluating the quality of the clusters, the reported approach achieved a cluster F-measure of 0.93. The approach outperformed that two baseline approaches in which no normalization was done or normalization was done when two entities had matching name mentions. The two approaches achieved cluster F-measures of 0.62 and 0.74 respectively.

For future work, implicit links between entities in the text can serve as the relational links that would enable the use of entity attributes in conjunction with relationships between entities. An important problem that has not been sufficiently explored is cross-lingual cross-document normalization. This problem would pose unique and interesting challenges. The described approach could be generalized to perform normalization of entities of different types across multilingual documents. Also, the normalization problem was treated as a classification problem. Examining the problem as a clustering (or alternatively an incremental clustering) problem might prove useful. Lastly, the effect of cross-document normalization should be examined on applications such as information extraction, information retrieval, and relationship and social network visualization.

References:

Bhattacharya I. and Getoor L. "A Latent Dirichlet Allocation Model for Entity Resolution." 6th SIAM Conference on Data Mining (SDM), Bethesda, USA, April 2006.

Chinchor N., Brown E., Ferro L., and Robinson P. "Named Entity Recognition Task Definition." MITRE, 1999.

Cohen W., Ravikumar P., and Fienberg S. E. "A Comparison of String Distance Metrics for Name-Matching Tasks." In Proceedings of the International Joint Conference on Artificial Intelligence, 2003.

Dozier C. and Zielund T. "Cross-document Co-Reference Resolution Applications for People in the Legal Domain." In 42nd Annual Meeting of the Association for Computational Linguistics, Reference Resolution Workshop, Barcelona, Spain. July 2004.

Fleischman M. B. and Hovy E. "Multi-Document Person Name Resolution." In 42nd Annual Meeting of the Association for Computational Linguistics, Reference Resolution Workshop, Barcelona, Spain. July 2004.

Ji H. and Grishman R. "Applying Coreference to Improve Name Recognition". In 42nd Annual Meeting of the Association for Computational Linguistics, Reference Resolution Workshop, Barcelona, Spain. July (2004).

Ji H. and Grishman R. "Improving Name Tagging by Reference Resolution and Relation Detection." ACL 2005

Joachims T. "Learning to Classify Text Using Support Vector Machines." Ph.D. Dissertation, Kluwer, (2002).

Joachims T. "Optimizing Search Engines Using Click-through Data." Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD), (2002).

Lee Y. S., Papineni K., Roukos S., Emam O., Hassan H. "Language Model Based Arabic Word Segmentation." In ACL 2003, pp. 399-406, (2003).

Li H., Srihari R. K., Niu C., and Li W. "Location Normalization for Information Extraction." Proceedings of the 19th international conference on Computational linguistics, pp. 1-7, 2002

Li H., Srihari R. K., Niu C., and Li W. "Location Normalization for Information Extraction." Proceedings of the sixth conference on applied natural language processing, 2000. pp. 247 – 254.

Mann G. S. and Yarowsky D. "Unsupervised Personal Name Disambiguation." Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003. pp. 33-40.

Maynard D., Tablan V., Ursu C., Cunningham H., and Wilks Y. "Named Entity Recognition from Diverse Text Types." Recent Advances in Natural Language Processing Conference, (2001).

Palmer D. D. and Day D. S. "A statistical Profile of the Named Entity Task". Proceedings of the fifth conference on Applied natural language processing, pp. 190-193, (1997).

R. Florian R., Hassan H., Ittycheriah A., Jing H., Kambhatla N., Luo X., Nicolov N., and Roukos S. "A Statistical Model for Multilingual Entity Detection and Tracking." In HLT-NAACL, 2004.

Rosell M., Kann V., and Litton J. E. "Comparing Comparisons: Document Clustering Evaluation Using Two Manual Classifications." In ICON 2004

Sekine S. "Named Entity: History and Future". Project notes, New York University, (2004).

Solorio T. "Improvement of Named Entity Tagging by Machine Learning." Ph.D. thesis, National Institute of Astrophysics, Optics and Electronics, Puebla, Mexico, September 2005.