Exploring the utility of coreference chains for improved identification of personal names

Andrea Glaser, Jonas Kuhn
Institute for Natural Language Processing
University of Stuttgart, Germany
andrea.glaser@ims.uni-stuttgart.de

Abstract

Identifying the real world entity that a proper name refers to is an important task in many NLP applications. Context plays an important role in disambiguating entities with the same names. In this paper, we discuss a dataset and experimental set-up that allows us to systematically explore the effects of different sizes and types of context in this disambiguation task. We create context by first identifying coreferent expressions in the document and then combining sentences these expressions occur in to one informative context. We apply different filters to obtain different levels of coreference-based context. Since hand-labeling a dataset of a decent size is expensive, we investigate the usefulness of an automatically created pseudo-ambiguity dataset. The results on this pseudo-ambiguity dataset show that using coreference-based context performs better than using a fixed window of context around the entity. The insights taken from the pseudo data experiments can be used to predict how the method works with real data. In our experiments on real data we obtain comparable results.

Keywords: named entity disambiguation, coreference, pseudo-ambiguity dataset

1. Introduction

Identifying the real world entity that a proper name refers to is an important task in many NLP applications such as web search, machine translation, and information retrieval. For example, the personal name *Michael Jackson* can be linked to a famous singer, a writer and beer expert, a Canadian actor, as well as many other people. An automatic tool helping with this task has to compare the context of a given usage of an ambiguous name with grounded contexts for the candidate referents (e.g., usages of the name including an explicit link to Wikipedia or some authority file).

In this paper, we discuss a dataset and experimental set-up that allows us to systematically explore the effects of different sizes and types of context in this disambiguation task. In particular, we approach the research question of how helpful it is to expand the relevant context in a linguistically informed way by adding the context of expressions from the same document that are (automatically) identified as coreferent. We choose several filter methods to obtain different sizes of context and use different levels of coreference-based context expansion.

One general problem is the lack of annotated datasets and hand-labeling a dataset of a decent size is expensive. For this reason we investigate the usefulness of an automatically created pseudo-ambiguity dataset. To create this dataset we choose similar entities with different names and extract documents that mention these entities. The names of these entities are then replaced by one artificial name that a classifier needs to disambiguate. The original names are used as the gold standard to evaluate the system. This pseudo-ambiguity dataset can be used to predict how our method works on real data.

In summary, we make the following contributions: (i) We show how different sizes of relevant context help improve disambiguation of personal names. (ii) We show how a large dataset for pseudo-ambiguity can be created automatically and discuss the possibilities of such a dataset. (iii) We create a smaller hand-labeled dataset based on our analysis of the pseudo-ambiguity dataset and show that when applying our method to this hand-labeled dataset we achieve similar results compared to applying it to pseudo data.

The paper is structured as follows. After discussing related work in Section 2, we will present our approach and how we create context in Section 3. Section 4 describes how we create the pseudo-ambiguity dataset. Section 5 gives an overview of the data we use and the classification method. Section 6 discusses the pseudo-ambiguity dataset and presents results for experiments on both pseudo and real data. In Sections 7 and 8 we give our conclusions and describe future work.

2. Related work

Mann and Yarowsky (2003) extract biographic facts (e.g., birth date, birth place, and occupation) from 1000 web pages. These biographic facts as well as the most relevant words in the document collection are used to disambiguate personal names with an unsupervised clustering technique. They create a dataset consisting of 28 pseudo-names, derived from combining 8 different people with similar backgrounds. However, they do not provide any detailed results of their experiments with this pseudo data, but only the overall disambiguation accuracy over all 28 pseudo-names. They also perform experiments on a hand-labeled dataset of real ambiguous persons; however, they only evaluate the two major sense clusters. In our study we will discuss the usefulness of pseudo data for this task as well as compare pseudo data and hand-labeled data directly.

Pedersen et al. (2005) identify significant bigrams as features and experiment with smaller and larger training and test scopes (5 and 20 bigrams, respectively) around the ambiguous name. They conduct their experiments on six pairs of pseudo-names (persons, organizations, nations, countries), but they do not apply their approach to real ambiguous data.

Much recent work has exploited Wikipedia and other knowledge bases as rich resources of information about named entities (Bunescu and Pasca, 2006; Shen et al., 2012). Cucerzan (2007) extracts different features from entity pages, redirecting pages, disambiguation pages, and list pages. He performs in-document coreference to map short surface forms to longer surface forms.

Cross-document coreference determines whether mentions of named entities in different documents refer to the same real world entity (Bagga and Baldwin, 1998; Fleischman and Hovy, 2004; Baron and Freedman, 2008). Similar to our work is (Bagga and Baldwin, 1998). They use within-document coreference to first identify all noun phrases that are coreferent with a given entity. They then create small document summaries with all sentences that contain these noun phrases and cluster the documents based on these summaries using a Vector Space Model. Our work is different in that we use several filters to create different summaries to find out whether using all sentences for the summary improves the disambiguation or introduce more noise. Their evaluation on a small hand-labeled corpus with only one ambiguous person is not exhaustive. In our paper we use pseudo data to conduct tests with several persons with different backgrounds first and then apply our method to real data.

Gooi and Allan (2004) base a large part of their work on (Bagga and Baldwin, 1998). They create the "Person X" corpus, a pseudo-ambiguity dataset where they replace occurrences of personal names of the form *firstname lastname* with *person-x*. However, they filter out cases that consist of only one word (e.g., *John*), while we also replace them and use them for creating context. Another difference is that they do not specifically choose persons with similar backgrounds, but use many different persons with a large variety of backgrounds.

3. Approach

The context a personal name occurs in provides helpful cues that help identify the real world entity. Many approaches experiment with a fixed window around the entity (for example *N* sentences, words, or bigrams to the left and right). This approach has two disadvantages. First, the immediate context might not contain relevant information about the entity but instead information about other entities, which adds noise to the context. Second, more relevant information that is located somewhere else in the document will be missed.

In our approach we assume that every sentence that contains the entity we want to identify conveys relevant information about it. The entity does not need to be mentioned explicitly as a proper noun but can also be substituted by a common noun or pronoun. We create our context by identifying all these relevant sentences in a given document and combining them to form one informative context. To find all relevant sentences that mention the entity, we perform coreference resolution. Coreference resolution is the task of determining whether two linguistic expressions (mentions) refer to the same real world entity or not. Two mentions are coreferent if they refer to the same entity, otherwise they are disreferent. All mentions in a document that refer to the same entity constitute a coreference chain.

(1) *(Ehud Barak)$_a$ told (President Clinton)$_b$ of (his)$_a$ plans for starting talks with (the Palestinians)$_c$. (He)$_a$ specifically discussed the 15-month target with (Clinton)$_b$, and (the president)$_b$ agreed to the urgency.*

Example (1) shows two sentences annotated with coreference information for mentions with the semantic class *person*. These mentions include noun phrases where the head is a proper noun (*Ehud Barak, President Clinton, the Palestinians, Clinton*), common noun (*the president*), or pronoun (*his, he*). The three coreference chains for this example are the following:

(a) *[Ehud Barak - his - he]*

(b) *[President Clinton - Clinton - the president]*

(c) *[the Palestinians]*

3.1. Context creation

To create context for an entity *e* we want to disambiguate, we perform the following steps:

1. The basis context is defined as the sentence *e* occurs in.
2. We build the coreference chain for *e* by determining all coreferent mentions of *e*.
3. For every mention *m* in the coreference chain, we extract the sentence *m* occurs in as a potential candidate sentence to be included in our relevant context.
4. The extracted sentences will then be added to the context depending on one of the following four filters:

- **nnp:** Use only sentences that contain the entity as a proper noun (e.g., *Ehud Barak, Barak*).
- **nnp+nn:** Use sentences that contain the entity as a proper noun or common noun (e.g., *Ehud Barak, Barak, the Prime Minister*).
- **nnp+prp:** Use sentences that contain the entity as a proper noun or pronoun (e.g., *Ehud Barak, Barak, he, his*).
- **nnp+nn+prp:** Use all sentences that contain the entity (e.g., *Ehud Barak, Barak, the Prime Minister, he, his*).

Figure 1 shows examples of different context sizes for identifying the personal name *Ehud Barak* in the first sentence. The original text is annotated with all mentions that are coreferent with the entity and their part of speech tags. The first two examples after the original text either take only the sentence the personal name occurs in as context (basis context) or add the previous and next sentence (if available) to the basis context. Since in this example the personal name occurs in the first sentence of the document, no previous sentence can be selected here. The next sentence is not about the entity, so it does not provide relevant information. The last four examples show how context is created with our filter methods. If the entity is mentioned several times

Original text

Israelis overwhelmingly elected (**Ehud Barak**)$_{1[nnp]}$, (the Labor Party leader)$_{1[nn]}$, as (prime minister)$_{1[nn]}$ on Monday. In a complete reshuffling of the political deck, Israelis also remade their parliament. (Barak)$_{1[nnp]}$ has the option to form a government without any ultra-Orthdox representation. By 3 a.m. Tuesday, (he)$_{1[prp]}$ was leading with 57 percent of the vote to Netanyahu's 42.8 percent. Within 35 minutes of the first television exit polls, Netanyahu not only conceded defeat, but stepped down as the leader of the conservative Likud Party. (The new prime minister)$_{1[nn]}$ now has 45 days in which to form a government.

Basis context

Israelis overwhelmingly elected Ehud Barak, the Labor Party leader, as prime minister on Monday.

Basis context and previous/next sentence

Israelis overwhelmingly elected Ehud Barak, the Labor Party leader, as prime minister on Monday. In a complete reshuffling of the political deck, Israelis also remade their parliament.

nnp filter

Israelis overwhelmingly elected Ehud Barak, the Labor Party leader, as prime minister on Monday. Barak has the option to form a government without any ultra-Orthdox representation.

nnp+nn filter

Israelis overwhelmingly elected Ehud Barak, the Labor Party leader, as prime minister on Monday. Barak has the option to form a government without any ultra-Orthdox representation. The new prime minister now has 45 days in which to form a government.

nnp+prp filter

Israelis overwhelmingly elected Ehud Barak, the Labor Party leader, as prime minister on Monday. Barak has the option to form a government without any ultra-Orthdox representation. By 3 a.m. Tuesday, he was leading with 57 percent of the vote to Netanyahu's 42.8 percent.

nnp+nn+prp filter

Israelis overwhelmingly elected Ehud Barak, the Labor Party leader, as prime minister on Monday. Barak has the option to form a government without any ultra-Orthdox representation. By 3 a.m. Tuesday, he was leading with 57 percent of the vote to Netanyahu's 42.8 percent. The new prime minister now has 45 days in which to form a government.

Figure 1: Examples of different context sizes.

in a sentence (e.g., as a proper noun and a common noun like in the first sentence), this sentence will be used only once.

We test different filters for two reasons. First, we want to find out the effect of different content sizes and whether every sentence that contains the entity adds useful information to the context. Second, since we do not have gold information for coreferent entities but determine coreference automatically, the coreference chain is not completely reliable and can contain wrongly classified entities. Classifying proper nouns correctly is easier because the string matching feature is quite reliable. However, pronouns always are more difficult to identify correctly. Adding a sentence to the context that is about another entity, adds more noise to the context.

4. Pseudo-ambiguity dataset

Our task of obtaining relevant context by exploiting coreference information to disambiguate personal names requires a dataset that is annotated with (i) disambiguation of named entities and (ii) coreference information. There exist some datasets for (i) that are hand-labeled, but most of them are small. Moreover, some documents in these collections mention the entity only once. Since we are interested in investigating how different context sizes influence the disambiguation, documents that only mention the entity once do not provide enough additional context. On the other hand, datasets with gold information about coreference chains also exist, but they do not contain named entity disambiguation.

Hand-labeling a dataset of a decent size is expensive. To overcome this problem we automatically create a pseudo-ambiguity dataset. Pseudo data has been used for various disambiguation tasks and was originally proposed for disambiguating word senses (Schütze, 1992; Gale et al., 1992).

Original texts

Israelis overwhelmingly elected **Ehud Barak**, the Labor Party leader, as prime minister on Monday. **Barak** has the option to form a government without any ultra-Orthdox representation.

Ariel Sharon has said he is determined to keep the city of Jerusalem in Israeli hands. **Sharon** was fired as defense minister in 1983 after a commission found him indirectly responsible for the 1982 massacre of more than 800 Palestinian refugees.

Replaced texts

Israelis overwhelmingly elected **EhAri BarShar**, the Labor Party leader, as prime minister on Monday. **BarShar** has the option to form a government without any ultra-Orthdox representation.

EhAri BarShar has said he is determined to keep the city of Jerusalem in Israeli hands. **BarShar** was fired as defense minister in 1983 after a commission found him indirectly responsible for the 1982 massacre of more than 800 Palestinian refugees.

Figure 2: Replacement with artificial pseudo-names.

For creating a pseudo-ambiguity dataset for personal names, we chose pairs of persons with different names but similar backgrounds; for example *Ehud Barak* and *Ariel Sharon* (Prime Ministers of Israel). We replace every occurrence of the last names *Barak* and *Sharon* with the artificial pseudo-name *BarShar* and respectively *EhuAri* for the first names *Ehud* and *Ariel*. An example of how this

	X	Y	# Docs	# Samples Barak	# Samples Sharon	# Samples Gore	# Samples Cheney	# Samples Milosevic	# Samples Kostunica
1	1	4	100	915	750	787	1214	683	821
2	1	4	200	1891	1649	1529	2286	1329	1521
3	1	4	300	2909	2820	2311	3199	-	-
4	1	2	100	696	523	405	649	517	533
5	1	3	100	877	611	589	917	609	736
6	1	4	100	915	750	787	1214	683	821
	X	Y	# Docs	# Samples Martin	# Samples Faulk	# Samples Williams (S)	# Samples Williams (R)		
7	1	2	100	932	793	426	414		

Table 1: Number of samples extracted from English Gigaword

replacement looks like can be found in Figure 2. To avoid unwanted ambiguity, we choose famous names for this task because all mentions in the data most likely refer to this person. The original names are used as gold labels for evaluating the system.

We first investigated OntoNotes 4.0[1], which is annotated with gold coreference information. However, the number of obtained samples (43-154 for each person) was too small to make a reliable statement about the usefulness of additional context. It showed that it is important to use a bigger dataset to obtain more accurate results.

To be more independent from existing datasets, we decided to annotate documents with coreference information automatically. To test the accuracy of such an automatic annotation system (see Section 5.2 for details about the system) we annotated the OntoNotes corpus automatically with it and compared the system output with the gold labels provided by OntoNotes. We found that the accuracy of the automatically predicted labels is quite high.

We test different methods for automatically extracting documents about a person from the collection. For example, if an entity is mentioned many times in a document then it is more likely that the document is about this entity and provides more relevant information.

The pseudo-ambiguity dataset also enables us to analyze minimum and optimal numbers of documents required to achieve reliable results or to see if more documents are needed. Based on these numbers we can then create a hand-labeled dataset for the task.

5. Experiments

5.1. Data

The English Gigaword corpus[2] (Graff and Cieri, 2003) comprises approximately 4.1 million documents of English newswire data from four international sources. The corpus is not annotated linguistically, i.e., the documents are available as raw text only.

For our experiments with pseudo-names we choose pairs of persons with similar backgrounds (profession and nationality) that have a relatively high frequency in the Gigaword corpus: *Ehud Barak/Ariel Sharon* (Israeli Prime Ministers), *Al Gore/Dick Cheney* (American Vice Presidents), *Slobodan Milosevic/Vojislav Kostunica* (Serbian/Yugoslav Presidents), and *Curtis Martin/Marshall Faulk* (American football players). For our tests on real data we use the two American football players named *Roy Williams* (one of them a safety, the other one a wide receiver).

We use the following heuristics to extract documents about a person automatically from the collection: the full name of the person occurs at least *X* times in the document and the last name occurs at least *Y* times. The higher these numbers are, the more relevant sentences about the entity can be used for the context, but the less documents will be extracted.

Table 1 shows the number of samples (proper noun mentions that we want to identify) that we extract from different numbers of documents per person when using different values for *X* and *Y*. Some persons occur less frequently in the corpus. In these cases we had to choose smaller numbers for *X* and *Y* to obtain enough documents.

5.2. Preprocessing

The Stanford CoreNLP system provides all necessary tools to annotate the extracted raw data, including part-of-speech tagging (Toutanova et al., 2003), named entity recognition (Finkel et al., 2005), and coreference resolution (Raghunathan et al., 2010; Lee et al., 2011; Lee et al., 2013; Recasens et al., 2013). The Stanford Deterministic Coreference Resolution System uses several sieves of features and aims at achieving high precision results, which provides reliable coreference chains. In our experiments we use all sieves to include pronoun matches.

5.3. Classification

We implement two baselines. The first baseline (00) uses the sentence that contains the entity with no further context. The second baseline (11) adds the previous and next sentences (if available) as additional context to the 00 baseline sentence. The idea behind the second baseline is that sentences that are close to the mentioned personal name are more likely to contain additional relevant information.

The input data from the baselines as well as the input data created according to the filter methods in Section 3.1 are represented as bag of words features (after removing stop words).

We use the machine learning software WEKA (Hall et al., 2009) to classify our data and implemented the following classifiers: (i) Naive Bayes (John and Langley,

[1] `http://catalog.ldc.upenn.edu/LDC2011T03`
[2] `http://catalog.ldc.upenn.edu/LDC2003T05`

Filter	100 docs NB	J48	SMO	200 docs NB	J48	SMO	300 docs NB	J48	SMO
00	**.764**	**.731**	**.789**	.691	.708	.784	.680	.720	.788
11	**.860**	.800	**.866**	.775	.786	.821	.765	**.803**	.838
nnp	**.977**	**.850**	.941	.918	.828	**.945**	.915	.846	.943
nnp+nn	**.967**	.836	.939	.915	.820	**.946**	.912	**.843**	.942
nnp+prp	**.969**	**.863**	.943	.905	.831	.952	.908	.812	**.956**
nnp+nn+prp	**.961**	**.873**	.943	.917	.834	**.953**	.916	.824	.951

Table 2: Comparison of different numbers of used documents (Barak/Sharon, X=1, Y=4)

Filter	Y=2 NB	J48	SMO	Y=3 NB	J48	SMO	Y=4 NB	J48	SMO
00	.666	.601	.686	.671	.673	.702	**.764**	**.698**	**.789**
11	.732	.665	.724	.741	.721	.766	**.860**	**.800**	**.866**
nnp	.842	.765	.860	.910	.781	.882	**.977**	**.850**	**.941**
nnp+nn	.836	.712	.863	.918	.785	.876	**.967**	**.836**	**.939**
nnp+prp	.845	.683	.824	.871	.855	.901	**.969**	**.863**	**.943**
nnp+nn+prp	.832	.654	.854	.866	.849	.904	**.961**	**.873**	**.943**

Table 3: Comparison of different automatic extraction methods (Barak/Sharon, 100 docs, X=1)

1995), (ii) J48 (Decision Tree C4.5 implementation, Quinlan (1993)), (iii) SMO (John Platt's sequential minimal optimization algorithm for training a support vector classifier, Platt (1998)).

We evaluate using 5-fold cross-validation. We report the weighted average F1-measure over both entities in each entity pair experiment for both pseudo and real data experiments.

6. Results and discussion

6.1. Effect of size

We investigated how different corpus sizes influence the results. This is useful because it can show if a large corpus is needed for a certain task or if a smaller one is sufficient. For this purpose we conducted the same experiments on 100, 200, and 300 (if available) documents for each entity. The number of samples for these experiments on documents extracted with *X*=1 and *Y*=4 is listed in lines 1-3 in Table 1. We use the entity pair *Barak/Sharon* to illustrate in Table 2 how results vary with different numbers of documents. Bold numbers show the best results for each classifier. Taking more documents does not always help improve the results. Naive Bayes and J48 (with some exceptions) both perform better on the smaller dataset of 100 documents. The SMO classifier generally benefits from more data. The results for the other pairs are similar: in many experiments with different entity pairs, using 100 documents achieved the best results. This shows that for our task a smaller dataset is sufficient. However, when the task changes or for example more noise and errors are filtered out, these experiments need to be run again to determine a new optimal size.

We conducted several experiments on the same numbers of documents that we extracted by varying the values *X* and *Y*. We kept *X*=1 (the full name has to occur at least one time), but set *Y* (the minimum number of occurrences for the last name only) to a value between 2 and 4. The number of samples for these experiments can be found in lines 4-6 in Table 1.

Table 3 shows classification results for the entity pair *Barak/Sharon* with 100 extracted documents in each experiment. Increasing the values for *X* and *Y* improves the results in all cases. One reason is that the more often an entity is mentioned in the document explicitly, the larger the context gets with more relevant sentences. The other reason is that if an entity occurs often in a document, the document is more likely to focus on this entity. This means extracted sentences contain more relevant information about the entity and do not only mention the entity as a side issue while actually talking about another entity.

6.2. Results on pseudo data

Table 4 shows the final results on the pseudo-name experiments for politicians (*Ehud Barak/Ariel Sharon*, *Al Gore/Dick Cheney*, and *Slobodan Milosevic/Vojislav Kostunica*). For these experiments we use the best setting as determined, i.e., *X*=1, *Y*=4, and 100 documents for each entity.

The first two lines show results for the two baselines. Adding the previous and next sentences to the context (11 baseline) helps improve the classification results because often more detailed information about an entity is mentioned in additional sentences.

The last four lines present the results achieved with our system which outperforms both baselines in all cases. One reason is that the contexts consist of more sentences that mention the entity in the document, which is usually larger than the context of the baselines. The second important reason is the contexts are likely to contain more relevant information about the entity, while the additional sentences in the 11 baseline might contain only little or no relevant information.

Filter	Barak/Sharon NB	J48	SMO	Gore/Cheney NB	J48	SMO	Milosevic/Kostunica NB	J48	SMO
00	.764	.731	.789	.825	.759	.837	.822	.796	.833
11	.783	.777	.792	.899	.865	.910	.901	.851	.929
nnp	**.977**	.850	.941	.928	**.930**	**.988**	.950	**.932**	.970
nnp+nn	.967	.836	.939	.928	**.930**	**.988**	.950	**.932**	.970
nnp+prp	.969	.863	**.943**	.933	.889	.981	.961	.904	**.972**
nnp+nn+prp	.961	**.873**	**.943**	**.937**	.890	.981	**.966**	.904	.962

Table 4: Results on pseudo data for politicians (100 docs, X=1, Y=4)

Filter	pseudo data Martin/Faulk NB	J48	SMO	real data Williams/Williams NB	J48	SMO
00	.731	.634	.711	.780	.706	.781
11	.798	.758	.787	.840	.819	.830
nnp	.854	.869	.902	**.858**	.835	.920
nnp+nn	.867	.867	.879	**.858**	.835	**.923**
nnp+prp	**.899**	**.900**	.874	.841	**.875**	.922
nnp+nn+prp	.875	.895	**.907**	.823	**.875**	.922

Table 5: Results on pseudo data and real data for sportsmen (100 docs, X=1, Y=2)

Bold values show the best results for each filter method and each classifier. There is no clear pattern of which is the best filter method. Instead, the best filter method varies from pair to pair. In some cases only adding sentences that mention the entity as a proper noun (nnp) yields the best results and adding more sentences that mention the entity as a common noun (nnp+nn) or pronoun (nnp+prp) worsens the results. The main problem lies in our data. Since we annotated the data with linguistic information automatically, it is likely that in the coreference resolution step some mentions are incorrectly classified as coreferent with another entity. Determining coreference for proper nouns is usually an easy task because the string matching feature is very reliable (except for cases where people with the same name occur in the same document, which was not the case in our experiments). However, the task of determining coreference for common nouns and pronouns is more difficult. If mentions are classified incorrectly, sentences that are not relevant to the entity will be added to the context. This adds more noise and makes it harder to classify.

These results show two things. First, adding more relevant context improves the classification results compared to using a specific window around the entity. Second, if no gold coreference information annotated by humans is available simply choosing the largest relevant context is not always the best method.

6.3. Results on real data

We created a small, hand-labeled dataset for the two American football players *Roy Williams* (one of them a safety, the other one a wide receiver) that we extracted by applying X=1 and Y=2. Larger values were not possible because both persons are not in the English Gigaword corpus very frequently and increasing the values did not give us enough documents. All extracted documents were manually checked if they were about one of these persons. Documents about other persons with the same name were discarded. The final corpus consists of 249 documents in total: 131 documents about the saftey and 118 documents about the wide receiver.

We choose the number of documents for the real data experiments based on our observations on our pseudo data. Since 100 documents often yield the best results in the pseudo data experiments, we also used 100 documents for each entity in the real data expriments. The number of samples can be found in line 7 in Table 1.

Table 5 compares classification results on pseudo data on the left side with real data on the right side. To make it more comparable to the real data, we used two other American football players (*Curtis Martin/Marshall Faulk*) for these pseudo data experiments.

Results on pseudo data for sportsmen are similar to results for politicians in that our system always performs better than the baselines but that there is no clear best method.

The results on the real data show comparable results to the pseudo data. Expanding the context with our different filters outperforms both baseline in all cases. However, the improvement when using Naive Bayes and J48 compared to the baseline is smaller than it was in most pseudo data experiments. When analyzing the data we discovered that while both players always played for different teams, they were both playing for the same team at different times. The team name is a very prominent feature when disambiguating these two football players, but the fact that the same team name was mentioned for both players in some documents, makes the disambiguation task more difficult.

7. Conclusion

We showed that for disambiguating personal names it is helpful to expand a basis context around the entity by first identifying coreferent expressions in the document and then adding the sentences these expressions occur in to one informative context. We applied different filters to add different levels of coreference-based context which all outperform two baselines that only use context around the personal name (the basis context or the basis context and previous/next sentences). We discuss that simply adding all possible context does not necessarily lead to the best results.

We conclude that pseudo data is a useful resource to test our proposed method without needing to manually annotate a large dataset of real data. It is easy and inexpensive to create large datasets to test different settings, such as optimizing the ideal corpus size and using different sizes of context. The insights taken from the pseudo data experiments can then be used to predict how the method works with real data. In our experiments on real data we obtained comparable results.

8. Future work

An important step to improve the results and make them more reliable is to reduce errors made in the automatic preprocessing, especially the coreference resolution task. To do this, we plan to test different coreference resolution system for their individual and also combined performance.

Another important issue is the fact that some properties of a person change over time, and some of them faster than others. For example they change their profession (e.g., many politicians hold different positions during their life) or their work place (e.g., football players change their teams). To optimize personal name identification, these time effects need to be taken into account in future work.

9. Acknowledgements

The work reported in this paper was supported by a Nuance Foundation Grant.

10. References

Bagga, A. and Baldwin, B. (1998). Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*, COLING '98, pages 79–85, Stroudsburg, PA, USA. Association for Computational Linguistics.

Baron, A. and Freedman, M. (2008). Who is who and what is what: Experiments in cross-document co-reference. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 274–283, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bunescu, R. and Pasca, M. (2006). Using encyclopedic knowledge for named entity disambiguation. In *Proceesings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '06, pages 9–16, Stroudsburg, PA, USA.

Cucerzan, S. (2007). Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '07, pages 708–716, Stroudsburg, PA, USA. Association for Computational Linguistics.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 363–370, Stroudsburg, PA, USA. Association for Computational Linguistics.

Fleischman, M. and Hovy, E. (2004). Multi-document person name resolution. In *Workshop on Reference Resolution and its Applications*, ACL '04, pages 1–8, Stroudsburg, PA, USA, July. Association for Computational Linguistics.

Gale, W. A., Church, K. W., and Yarowsky, D. (1992). Work on statistical methods for word sense disambiguation. In Goldman, R., Norvig, P., Charniak, E., and Gale, B., editors, *Working Notes of the AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, pages 54–60, Menlo Park, CA. AAAI Press.

Gooi, C. H. and Allan, J. (2004). Cross-Document Coreference on a Large Scale Corpus. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, HLT-NAACL '04, pages 9–16, Stroudsburg, PA, USA, May. Association for Computational Linguistics.

Graff, D. and Cieri, C. (2003). English Gigaword. Linguistic Data Consortium, Philadelphia.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: An update. *SIGKDD Explorations*, 11(1):10–18.

John, G. H. and Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Mateo. Morgan Kaufmann.

Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., and Jurafsky, D. (2011). Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task*, CoNLL Shared Task '11, pages 28–34, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lee, H., Chang, A., Peirsman, Y., Chambers, N., Surdeanu, M., and Jurafsky, D. (2013). Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4):885–916.

Mann, G. S. and Yarowsky, D. (2003). Unsupervised personal name disambiguation. In *Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CoNLL '03, pages 33–40, Stroudsburg, PA, USA. Association for Computational Linguistics.

Pedersen, T., Purandare, A., and Kulkarni, A. (2005). Name discrimination by clustering similar contexts. In *Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing '05, pages 226–237, Berlin, Heidelberg.

Springer-Verlag.

Platt, J. C. (1998). Fast training of support vector machines using sequential minimal optimization. In Schölkopf, B., Burges, C. J. C., and Smola, A. J., editors, *Advances in Kernel Methods - Support Vector Learning*, pages 185–208. MIT Press, Cambridge, MA, USA.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., and Manning, C. (2010). A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 492–501, Stroudsburg, PA, USA. Association for Computational Linguistics.

Recasens, M., de Marneffe, M.-C., and Potts, C. (2013). The life and death of discourse entities: Identifying singleton mentions. In *Proceedings of 2013 the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, HLT-NAACL, pages 627–633. The Association for Computational Linguistics.

Schütze, H. (1992). Context space. In Goldman, R., Norvig, P., Charniak, E., and Gale, B., editors, *Working Notes of the AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, pages 113–120, Menlo Park, CA. AAAI Press.

Shen, W., Wang, J., Luo, P., and Wang, M. (2012). Linden: linking named entities with knowledge base via semantic knowledge. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 449–458, New York, NY, USA. ACM.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.