Web People Search Disambiguation using Language Model Techniques

Juan Martinez-Romo
Dpto. Lenguajes y Sistemas Informáticos
UNED
28040 Madrid, Spain
juaner@lsi.uned.es

Lourdes Araujo
Dpto. Lenguajes y Sistemas Informáticos
UNED
28040 Madrid, Spain
lurdes@lsi.uned.es

ABSTRACT

In this paper we describe our participation in Web People Search Clustering task. We present a new methodology based on language models to improve Web People Search disambiguation. In particular we introduce two different approaches: One of them uses alternative weighting functions to represent a document and it apply a classical clustering algorithm. The second approach uses two sources of occupational information as reference collections and it applies an heuristic-based technique in order to resolve the number of different identities. Moreover, we have studied the impact in results of using stemming and a variant of Interpolated Aggregate Smoothing applied to language models.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; I.2.7 [**Computing Methodologies**]: Natural Language Processing; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

General Terms

Algorithms, Experimentation, Measurement

Keywords

Web People Search, Clustering, Language Model approach

1. INTRODUCTION

Nowadays, Internet users do not only resort to the Web to search information as it has happened until recently, but they actively have participated by editing and publishing Web content several years ago. Thanks to search engines, free hosting sites and social networks systems is relatively easy to find information about a person online. Moreover, in recent years, there exists a tendency to publish on the Web a lot of personal information introducing new multimedia resources such as images, podcasts or videos, for instance in Web sites such as Fotolog[1], MySpace[2], etc. All these data can be found on the Internet are very useful when you need to find information about a person, but in many cases the names of individuals are shared and you can get thousands of pages belonging to different persons with the same name.

In this way the Web People Search (WePS) task at SemEval 2007[2] came up with the goal of disambiguate person names in a Web searching scenario. This year *WePS* task[3] continues with the same objective, but with a different dataset. This task has proposed to provide participants with a set of documents retrieved from a search engine. In this case, a query was formed by a person name and the final objective was to carry out a document clustering in which each set of documents clearly belong to a person identity.

In this paper we propose a new methodology based on language models to improve Web People Search disambiguation. Language models[11] are probabilistic methods that have been previously used successfully in areas of machine translation, part-of-speech tagging, information retrieval, Web Spam and even in some previous works for this task[4]. Statistical language models have been developed to capture linguistic features hidden in texts, such as the probability of words or word sequences in a language.

Specifically, we have used language modeling techniques to represent a document with the extracted terms that are most relevant. In addition to the documents retrieved by a search engine that are provided by the WePS task dataset, we have created two collections with documents that represent hundreds of occupations. Our main objective in this task has been to identify every person in a document with a occupation in order to automatically distinguish among different identities. Finally we have used clustering techniques and a set of heuristics based on the similarity of a document to a certain profession in order to optimize the performance of our system.

Previous works have used rich features in order to improve the disambiguation performance as Chen et al.[5] who extend token-based information to a web corpus, they also include some noun phrase-based information. Popescu et al.[12] used parsers and Named Entities Recognition (NER) techniques, and also they tried to recognize some terms denoting occupations. Elmacioglu et al.[7] extracted several sources of information from Web documents (tokens, named entities, etc.) and used them in the similarity computation of the clustering algorithm. Saggion[13] uses an agglomerative clustering approach to group documents referring to the same entity. This approach uses vector representations created by summarization and semantic tagging analysis com-

[1]http://www.fotolog.com/
[2]http://www.myspace.com/

WWW2009, April 20-24, 2009, Madrid, Spain.
.

ponents. Closest to our research is the work by Balog et al.[4] that adapted language model techniques in WEPS task. Specifically, they used single pass clustering to automatically assign pages to clusters, and probabilistic latent semantic analysis based on term co-occurrence. Moreover, they represented each document considering the title, snippet and body text.

The remaining of the paper proceeds as follows: section 2 describes the language model techniques used in experiments and the set of proposed features; section 3 enumerates the datasets features and the process of crawling; section 4 is devoted to describe the methodology we have followed to disambiguate person names and to carry out the clustering tasks; section 5 presents the experiments resulting of the previous methodology, as well as the results of applying it to WePS dataset; Finally, section 6 draws the main conclusions of this work.

2. LANGUAGE MODELS AND FEATURES

One of the main approaches to query expansion is based on studying the difference between the term distribution in the whole collection and in the subsets of documents that can be relevant for the query. One would expect that terms with little informative content have a similar distribution in any document of the collection. On the contrary, terms representative of a page or document are expected to be more frequent in that page than in other subsets of the collection considered.

One of the most successful methods based on term distribution analysis uses the concept of Kullback-Liebler Divergence[6] to compute the divergence between the probability distributions of terms in the whole collection and the particular documents considered. The most likely terms to expand the query are those with a high probability in the document which is the source of terms and low probability in the whole collection. For the term t this divergence is:

$$KLD(D_1||D_2) = \sum_{t \in D_1} P_{D_1}(t) log \frac{P_{D_1}(t)}{P_{D_2}(t)} \quad (1)$$

where $P_{D_1}(t)$ is the probability of the term t in the reference document, and $P_{D_2}(t)$ is the probability of the term t in the candidate document.

Computing this measure requires a reference collection of documents. The relation between this reference collection and the analysed document, is an import factor in the results obtained with this approach. Since general web pages are very different to each other, and can refer to any topic, it is not easy to find an appropriate reference collection. Obviously we can not use the whole web as reference collection, so we have assembled a collection of Web documents from a public Web directory. This collection is the result of a crawling process on a random set of Urls from the DMOZ Open Directory Project (ODP)[3]. The whole set is around 4.5 million of sites, but in this work we have only used a set of two million of sites randomly chosen. We set the crawling depth to zero, so just a document has been retrieved from each site (homepage site).

Moreover, it is well-known that there exist smoothing techniques for improving performance of language models. Specifically in this work we used a variant of Interpolated Aggregate Smoothing[15]. In this approach, probabilities are estimated using maximum likelihood models and smoothed with Jelinek-Mercer smoothing. We estimate the probabilities using maximum likelihood and smooth using a general probability model of terms from DMOZ collection previously defined. By using Interpolated Aggregate Smoothing, probabilities have the form:

$$p(t|D) = \lambda \cdot p(t|D) + (1 - \lambda) \cdot p(t|C) \quad (2)$$

where $p(t|D)$ is the probability of the term t in the considered document, and $p(t|C)$ is the probability of the term t in the whole collection.

3. REFERENCE COLLECTIONS

One of two algorithms presented in this work is based on the assumption that many person pages which may be recovered by a search engine correspond to a professional profile. Furthermore, other pages such as personal websites, pages in social networks, etc. often refer to the profession which a person perform as part of the narrative of his life. Even a personal page that refers to leisure time of a certain person might be related to an occupation, for instance: photographer, soccer player, singer.

For this reason we have created a collection of documents that represent hundreds of occupations. Our main goal in this task has been to assign each document that refers to a person to a profession in order to distinguish different identities. Our reasoning is based on the assumption that if we analyse a document and we manage to discover an underlying profession of this person or some task performed in his leisure time, we can disambiguate his identity in two documents. Finally we have used clustering techniques and a set of heuristics based on the similarity of a document and a profession in order to improve the performance of our system.

To study the impact of this factor on the results we have used two publicly available sources of occupational information:

- Wikipedia. This free encyclopedia has available a list of occupations[4] with over 670 documents. As it is well-know, these documents have a large description of every occupation and also there exists a list of references, external links and links to the Wikipedia.
- Onet. This Web site[5] is the US primary source of occupational information, and it was developed for the *U.S. Department of Labor*. ONET is a database that contains well researched job descriptions and other job information for jobs that exist in the *United States* (both government and private). In this site it can be found different classifications by topics as Keywords, Career

[3]http://www.dmoz.org

[4]http://en.wikipedia.org/wiki/List_of_occupations

[5]http://www.onetcenter.org/

Cluster, Job Family, etc. Every document has a list of information about an occupation as tasks, tools, technology, knowledge, skills, etc.

For building these collections we have fulfilled a crawling process and we have recovered all documents from each Web site. Every document of these occupations has been indexed by filtering stopwords extracted from a public list in the University of Glasgow[6]. For indexing tasks we used Lucene[8], which is a source information retrieval library. In addition, we have decided to analyze the impact on results of using stemming[10]. For that, we have created two versions of each collection, one by using the Stemming algorithm by Porter, which is available at the Snowball Web site[7]. Therefore, we have four collections with the following features: (i) Wikipedia: 671 documents and 65,909 terms, (ii) Wikipedia Stem: 671 documents and 51,225 terms, (iii) Onet: 950 documents and 20,444 terms, (iv) Onet Stem: 950 documents and 15,424 terms.

4. METHODOLOGY

We present two methods based on language modeling techniques with the aim of improving the clustering task. Both methods uses language models for terminology extraction from text, although each method use different collections and every collection is used in a different way.

4.1 Heuristic-based approach

The method we present uses language models for terminology extraction from text and is based on a heuristic approach to determine the number of identities that can be found for each person. The following process is applied to every person in the WePS dataset. First of all documents are indexed by filtering stopwords. Once we have all documents indexed, a set of statistics are extracted that will be used by language models later. Specifically, we extract the number of documents and the total frequency of the collection. Afterwards, for every document are extracted a set of relevant terms. For this task we applied Kullback-Liebler Divergence (KLD), using all documents previously indexed as reference collection.

Concerning most relevant terms, we filter those that: (i) are numbers, (ii) have less than 3 characters, (iii) have an idf[10][8] lower than 2.5, or (iv) have non-alphanumeric symbols. These terms are used to form a query by concatenating of them all. After several experiments, we set to 25 the number of terms that represent each document. Next, we perform a query in the collection of occupations, and then we obtain a ranking of the most relevant occupations to this current document. Note that the Lucene similarity formula used in ranking function is motivated by the cosine-distance and based on vector space model[10]. From this ranking is recovered the first hit and it is awarded a vote. After repeating the same process for each document, it is established a ranking with most voted professions. At this point it is applied a heuristic to determine the number of identities for this person, according to the length of the ranking and the number of votes from professions in the top hits.

After several experiments, we decided to select only those occupations that get more than two votes, and we built a new index including only these selected occupations. Finally, a new query is performed for each document by using this sub-collection in order to assign every document to first hit in the ranking.

4.2 Clustering algorithm

This method is composed of two phases, a first phase in which documents representation is carried out and a second phase in which a classical clustering algorithm is applied.

As part of the first phase, all documents are indexed by filtering stopwords and several statistics are computed (number of documents and total frequency) that will later be used to define the language models. From this collection is extracted a vocabulary with the most relevant terms by using language models in this process. Specifically we applied Kullback-Liebler Divergence for terminology extraction, by considering the index with all documents of a person as a single document, and DMOZ collection introduced in section 2 as reference collection. Vocabulary size is a configurable parameter that also depends on the number of documents and their length (terms). To represent a document we used a vector in which every component represents the weight of a vocabulary term. We use three weight functions based on plain text of Web pages:

- **TF-IDF**. Combination of weights of a term t in a document d is given by:

$$TF - IDF(d,t) = TF(d,t) \times IDF(t) \qquad (3)$$

  This weight is a statistical measure used to evaluate how important a word in a document is in a collection or corpus[14].

- **KLD**. Weight of a term t in a document d is defined by equation (2). Probability is estimated using maximum likelihood models and smoothed with Jelinek-Mercer smoothing.

- **Cosine-similarity**. In this case representation of a document is not given by a vector of vocabulary terms, but a vector with the occupations in the collection. The weighting function is based on cosine-based similarity of the document and each occupation.

In the second phase, we apply a classical clustering algorithm using the software *Cluto*[9]. We have used a partitional clustering algorithm, specifically we have used the *Direct* algorithm combined with a cosine-based similarity function.

5. EXPERIMENTS AND RESULTS

We carried out several experiments to determine the best parameters for two defined methods, such as: (i) number of terms to represent a document, (ii) minimum number of votes to filter occupations, (iii) size of the vocabulary, (iv) use of Stemming or (v) weighting function. Best results were

[6]http://ir.dcs.gla.ac.uk/resources/linguistic_utils/
[7]http://snowball.tartarus.org/
[8]Inverse Document Frequency

Table of results of the clustering task								
Run	BEP	BER	F_0.5_BEPBER	F_0.2_BEPBER	P	IP	F_0.5_PIP	F_0.2_PIP
UNED_3	0.66	0.39	0.40	0.38	0.71	0.48	0.51	0.48
UNED_1	0.68	0.37	0.39	0.37	0.73	0.44	0.48	0.45
UNED_4	0.59	0.43	0.39	0.39	0.66	0.51	0.49	0.48
UNED_5	0.64	0.38	0.39	0.37	0.69	0.48	0.50	0.47
UNED_2	0.66	0.36	0.37	0.35	0.72	0.43	0.47	0.44

Table 1: Table of results of the clustering task ranked by *FMeasure_0.5_BEP-BER*

obtained applying the Heuristic-based approach, therefore the submit runs to participate in the Clustering task were created using this method. Specifically, we combined two sources of occupational information by applying stemming and not applying stemming respectively. In addition, we included a run in which selection of the number of occupations was done by hand, and showing to a person the votes ranking.

Table 1 reports results of the clustering task. Runs are sorted according to F-measure of B-Cubed Precision and Recall set $\alpha = 0.5$. These clustering evaluation metrics are described in [1]. The way in which runs have been obtained are described below.

- **UNED_1**. ONET was used as reference collection and it was not applied stemming.
- **UNED_2**. ONET was used as reference collection and it was applied stemming during indexing and query formulation process.
- **UNED_3**. Wikipedia was used as reference collection and it was not applied stemming.
- **UNED_4**. Wikipedia was used as reference collection and it was not applied stemming. Optimal selection of number of occupations was manually adjusted.
- **UNED_5**. Wikipedia was used as reference collection and it was applied stemming during indexing and query formulation process.

From results we can say that stemming techniques do not improve neither the precision nor the recall. For this reason runs *UNED_5* and *UNED_2* have obtained the lowest score. Maybe the problem is the use of stemming because when we used it, obtained results are more ambiguous and the system effectiveness is reduced.

Concerning the use of different collections, runs which had employed Wikipedia (*UNED_3*, *UNED_4* and *UNED_5*) have got a higher score. *ONET* collection describes occupations in a more heterogeneous way, by generating a wider context and establishing relationships with other professions. However, *Wikipedia* describes the occupations in a deeper way and, it has therefore a much larger and specific vocabulary for every occupation. Although *ONET* offers a more complete description of occupations, its vocabulary is shared with other related occupations and this causes a less efficient performance in the disambiguation of person identities.

In *UNED_4*, optimal selection of number of occupations was manually adjusted, but it has gained a lower score than runs *UNED_3* and *UNED_1* (three runs used Wikipedia as reference collection). Therefore, it can be said that the heuristic used to determine the number of clusters has worked correctly, even improving the results of manual selection.

If we pay attention to the precision (P) in the results, it can be shown that runs that used *ONET* (*UNED_1* and and *UNED_2*) have obtained a higher precision that runs using *Wikipedia*. However, if we look at recall (R), we can notice that runs using *Wikipedia* had a higher value that runs using *ONET*. These results could be due to the size of collections: *Wikipedia* has 671 occupations while *ONET* has 950. In this way, as *ONET* has a greater number of occupations, ambiguity is smaller among different identities, because it distinguishes more efficiently among related occupations.

Other clustering evaluation metrics are purity (P) and inverse purity (IP). In Table1 is shown that runs used *ONET* have a higher purity that runs using *Wikipedia*. However, if we look at the inverse purity, it can be seen that runs using *Wikipedia* has obtained a higher value.

6. CONCLUSIONS

Every day, people have a greater interest for both, publishing his life on Internet, and knowing about lives of others. Moreover, from a professional point of view, every day there exist more resources to obtain information about a person. Even so, these systems still have high levels of ambiguity in their results, mixing documents, pictures and quotes from different people. In response to this unresolved problem, it has appeared the task in which we participate this year. In this way the Web People Search (WePS) task came up with the goal of disambiguate person names in a Web searching scenario.

We describe our participation in WePS task and we present a new methodology based on language models to improve WePS disambiguation. In particular we introduce two different approaches, one of these uses alternative weighting functions to represent a document and it applies a classical clustering algorithm, but we do not submit any run using this approach because we obtained worse results than with the presented methods. The second approach uses two sources

of occupational information as reference collections and it applies an heuristic-based technique in order to resolve the number of different identities. Specifically we have built two collections by indexing two lists of occupations from ONET and Wikipedia. Moreover, we have studied the impact in results of using stemming and a variant of Interpolated Aggregate Smoothing applied to language models (LM). These LMs are estimated using Jelinek-Mercer smoothing.

According to the clustering evaluation metric used in this task, best results were obtained by runs that used Wikipedia as reference collection and those that do not use stemming techniques. In this way, our best run has got in task a $FMeasure\_0.5\_BEP - BER = 0.40$.

In future works we would like to built a collection with a higher number of occupations, maybe by merging ONET and Wikipedia collections and including new occupations more specific, i.e. *tennis player* or *saxophonist*. Also, we would like to analyse new features to be used for computing the similarity and we will try to improve the performance of our clustering algorithm.

7. ACKNOWLEDGMENTS

Partially supported by the Spanish Ministry of Science and Innovation within the project QEAVis-Catiex (TIN2007-67581-C02-01) and the Regional Government of Madrid under the Research Network MAVIR (S-0505/TIC-0267).

8. REFERENCES

[1] E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, pages 1–26.

[2] J. Artiles, J. Gonzalo, and S. Sekine. The semeval-2007 weps evaluation: Establishing a benchmark for the web people search task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 64–69, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[3] J. Artiles, J. Gonzalo, and S. Sekine. Weps 2 evaluation campaign: overview of the web people search clustering task. In *Proceedings of the Second Web People Search Evaluation Workshop (WePS 2009)*, Madrid, Spain, April 2009. 18th WWW Conference.

[4] K. Balog, L. Azzopardi, and M. de Rijke. Uva: Language modeling techniques for web people search. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 468–471, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[5] Y. Chen and J. H. Martin. Cu-comsem: Exploring rich features for unsupervised web personal name disambiguation. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 125–128, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[6] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley-Interscience, New York, NY, USA, 1991.

[7] E. Elmacioglu, Y. F. Tan, S. Yan, M.-Y. Kan, and D. Lee. Psnus: Web people name disambiguation by simple clustering with rich features. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 268–271, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[8] O. Gospodnetic and E. Hatcher. *Lucene in Action*. Manning, 2004.

[9] G. Karypis. CLUTO-A Clustering Toolkit, 2002.

[10] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[11] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, New York, NY, USA, 1998. ACM.

[12] O. Popescu and B. Magnini. Irst-bp: Web people search using name entities. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 195–198, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[13] H. Saggion. Shef: Semantic tagging and summarization techniques applied to cross-document coreference. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 292–295, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[14] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[15] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214, 2004.