Hypergeometric Language Model and Zipf-Like Scoring Function for Web Document Similarity Retrieval

Felipe Bravo-Marquez, Gaston L'Huillier, Sebastián A. Ríos, and Juan D. Velásquez

University of Chile, Department of Industrial Engineering, Santiago, Chile
{fbravo,glhuilli}@dcc.uchile.cl, {srios,jvelasqu}@dii.uchile.cl

Abstract. The retrieval of similar documents in the Web from a given document is different in many aspects from information retrieval based on queries generated by regular search engine users. In this work, a new method is proposed for Web similarity document retrieval based on generative language models and meta search engines. Probabilistic language models are used as a random query generator for the given document. Queries are submitted to a customizable set of Web search engines. Once all results obtained are gathered, its evaluation is determined by a proposed scoring function based on the Zipf law. Results obtained showed that the proposed methodology for query generation and scoring procedure solves the problem with acceptable levels of precision.

1 Introduction

Classic Web search engines have been developed aiming to solve information requirements from users. As proposed in [5], Web search queries can be grouped into three categories: Informational queries, navigational queries, and transactional queries. In [4] a different information requirement is described. We called it, the Web document similarity retrieval problem (WDSRP). This consists of retrieving the most similar documents from the Web using as input a given document instead of a query. Solutions to WDSRP could be applied for plagiarism detection, document impact analysis, or as related ideas retrieval tool. Also, a variation of this problem is known as the Near-Duplicate Detection [3].

Web search engines could perfectly solve the WDSRP allowing users to send complete documents as inputs. However, search engines allow a maximum query length, because long queries take a huge computation time, are hard to cache, and are usually composed of many non relevant terms.

The main contribution of this work is to solve the WDSRP using a probabilistic language model for query generation. Also, a meta search scoring function based on Zipf law is proposed. This function scores considering just the information responded by each search engine, mainly the rankings of retrieved results. This approach aims to avoid costly text processing algorithms over the responded documents. Furthermore, we performed successful experiments in a real world application with promising results.

E. Chavez and S. Lonardi (Eds.): SPIRE 2010, LNCS 6393, pp. 304–309, 2010.

2 Previous Work

As described in [7], meta-search engines provide a single unified interface, where a user enters a specific query, the engine forwards it in parallel to a given list of search engines, and results are gathered and ranked into a single list.

The WDSRP has been studied by different researchers [4,9]. These works propose fingerprinting techniques for document representation. In both cases the fingerprint is composed by sentences used as queries. The queries are submitted into a meta-search engine for retrieving an extended list of similar candidate documents.

On the one hand, in [9] document snippets are retrieved and compared with the given document using cosine similarity from the vector space model. On the other hand, Pereira and Ziviani propose in [4] to retrieve the complete text from each Web document, and compare them using text comparison strategies, like Patricia trees and Shingles method.

3 Hypergeometric Language Model for Query Generation

As stated by [5], given a document D, from which a vocabulary V can be extracted, a language model M_D from D is a function that maps a probability measure over strings drawn from V. Language models are used as ranking functions in information retrieval, estimating the probability of generating a query q given a document language model M_D, i.e. $P(q|M_D)$, using a generative expression for the ranking function.

In our query generation task, the probabilistic distribution from the language model is used as a randomized term extraction procedure. The reason for using randomized term permutations, is that similar documents from D do not necessarily contain words in the same order. Furthermore, as a strong but realistic assumption, search engines, where queries will be submitted, treat user natural text queries following a bag of words property [1].

The Hypergeometric Language Model (HLM) is a proposed extension of language models inspired in the multivalued hypergeometric distribution [2], which provides a non replacement property. This means that terms are extracted one by one without replacement. The property is based on the hypothesis that a new term gives more information to a search engine than a repeated term in the generated query, considering that search engines allow a maximum length of input queries.

Consider that the extracted vocabulary is determined by the expression $V = \{t_1, \ldots, t_m\}$, where each term t_i in the document has an assigned positive value w_i stored in vector $\overrightarrow{w} = \{w_1, \ldots, w_m\}$. These values can be determined by several weighting approaches, like Boolean, *tf*, *tf-idf*, among others [8].

A generated query q can be modeled as a list of term pointers extracted from a given vocabulary V, defined by $q = s_1, \ldots s_n$, where each term pointer $s_j \in q$ is an integer taking values in $\{1 \ldots m\} \in V$.

By using the chain rule of probabilities, the probability of generating the query q from a language model M_D, can be defined as, $P(q|M_D) = P(s_1|M_D) \cdot$

$\ldots P(s_n|s_1,\ldots,s_{n-1},M_D)$ and the conditional distribution of extracting the token s_j given an accumulated generated query $q = s_1,\ldots s_n$ and the language model M_D, is determined by,

$$\hat{P}(s_k|q,M_D) = \begin{cases} 0 & \text{if } \exists s_j \in q, s_k = s_j \\ \frac{w_{s_k}}{||\overrightarrow{w}||_1 - \sum_{j=1}^{n} w_{s_j}} & \text{otherwise} \end{cases} \quad (1)$$

HLM also provides a random query generator function. This function generates a sequence of terms using equation 1, giving a higher probability of occurrence to the most relevant terms ranked with the given weighting approach. The function models the term extraction with a multinomial distribution parametrized by $(\frac{\overrightarrow{w}}{||\overrightarrow{w}||_1})$. The without replacement property is modeled by reconstructing the multinomial distribution after each extraction by reducing the dimensionality of $\overrightarrow{w}$ and V by removing the extracted term dimension. Finally, the number of terms extracted is defined by a query *length* parameter.

4 Meta Search Engine and Zipf-Like Scoring Function for Generated Queries

As the coverage of the Web is potentiated by using a set of search engines S, the document retrieval is proposed by the union of the indexed documents presented in each of the search engines used. Assuming that document D is represented by a set of queries Q, generated by HLM (section 3), each query $q \in Q$ is mapped to be submitted to a search engine $s \in S$.

A *queryAnswer* ω is defined as a tuple (s,q,r) , where r represents the ranking assigned by search engine s for query q. Each *queryAnswer* $\omega_{s,q,r}$ will point to a particular document. In our work, all queries originate from the same language model. The hypothesis is that if the set of Web search inverted indexes contains documents with a higher similarity to the given document D, these documents should be founded by many queries in the top of their ranks, and more than once.

After retrieving the set of *queryAnswer* ω objects, they are grouped into *metaAnswer* objects. A *metaAnswer* Ω ($|\Omega| \geq 1$) is a set of ω, where each element points to the same URL or Web document.

Finally, all *metaAnswer* objects are scored and ranked by a proposed approach, based on the Zipf-like distribution function, described below.

The Zipf law, proposed in [10], has been used in the natural language community for the analysis of term frequencies in documents. As stated by [6], if f denotes the popularity of an item and r denotes its relative rank, then f and r are related as $f = \frac{c}{r^{\beta}}$, where c is a constant and $\beta > 0$. If $\beta = 1$, then f follows exactly the Zipf law, otherwise, is it said to be Zipf-like.

In [6], the Web popularity is modeled as the Zipf law, where the relative frequency with which Web pages are requested for the r^{th} most popular Web page is proportional to $1/r$. Furthermore, in this work we propose to model the relevance of a given *queryAnswer* as a Zipf-like distribution, where the relevance of results presented in a Web search engine are inversely related to their rankings.

All queries are generated by the same language model and have an underlying search intention. If a specific *queryAnswer* appears more than once, the probability that the pointed document is related to the document from which the query was generated increases. Thus, the scoring strategy for a *metaAnswer* Ω is expressed by

$$ZipfLikeScore(\Omega) = \frac{1}{|Q|} \sum_{\omega \in \Omega} \frac{c_s}{r^{\beta_s}} \tag{2}$$

where $c_s \in [0, 1]$ is a constant which represents the average relevance of the best response of a Web search engine s, and β_s represents the decay factor of the results' relevance while the amount of retrieved results increases. With this score measure, we are estimating the relevance of the *queryAnswer* using its ranking and the reliability of search engine results. The score is normalized by the number of queries requested, in order to represent the score by a real value $\in [0, 1]$.

5 Experiments

According to the previously described procedures, a prototype using a term frequency weighting approach and a Spanish stopwords' list was implemented. The prototype allows the client to insert a text without length constraints. However, if a whole document is considered as input, this could increase the number of non relevant results retrieved, because of the randomized query generation process. Therefore, we recommend to use a single paragraph instead of a whole document since it is a self-contained independent information unit.

A hand-crafted set of paragraphs were extracted from a sample of Web documents. For this, a total number of 160 paragraphs were selected from different Spanish written Web sites. The respective URL was stored and classified into three different document types: bibliographic documents or school essays (type 1), blog entries or personal Web pages (type 2), and news (type 3). The distribution of the paragraph types was 77, 53, and 30 respectively. The selected search engines for the experiments were Google, Yahoo!, and Bing. The parameters used for each search engine for the query generation and the ranking procedures were the number of queries generated per paragraph, c_s and β_s, whose assigned values were $(3, 0.95, 0.5)$ for Google, and $(2, 0.93, 0.5)$ for Yahoo! and Bing respectively. Finally the term length of each query was assigned to 13. These values have been assigned by inspection and their formal estimation has been intentionally omitted.

After this, paragraphs were sent to the system as input. The top 15 answers from each paragraph were manually reviewed and classified as relevant or non relevant results (2400 Web documents). The criteria of labeling an answer as relevant was defined that the retrieved document must contain the given paragraph exactly.

The goal of this experiment is to measure the effectiveness of the model at satisfying user information needs. In this case, those needs are related to the WDSRP. The criteria of measuring the relevance of the results is the number of

documents containing exactly the given paragraph (DEPs). The selected evaluation measure was the *precision at k*, which is defined as the number of DEPs retrieved in the top k results divided by the number of documents retrieved in top k results.

6 Results and Discussion

Fig. 1 shows the *precision at k* for results retrieved by the whole set of paragraphs associated with their document types. It is easy to see that the *precision at k* differs with each type of document.

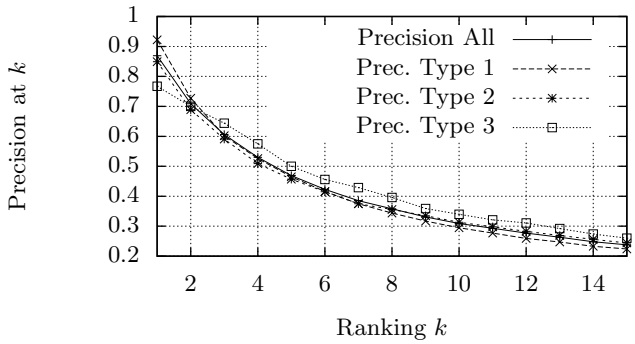

Fig. 1. Precision at k for types of documents

Firstly, type 1 has higher precision at the first values of k than the other types. This is because bibliographic documents are often founded in popular collections like Wikipedia, which are indexed by most Web search engines and usually ranked on top. In this case, a Web document will appear as result for most of generated queries. Secondly, type 2 documents are not as popular as type 1. In this case, blog entries or personal pages are hardly indexed by all Web search engines. Finally, we can observe a lower precision for the first ranked results of type 3 documents. However, a slow decline of the *precision at k* is presented. That is because news contents are repeated in many different Web pages, through the increasing use of content aggregators, so is possible to find a high number of relevant results lower ranked documents.

7 Conclusions

To the best of our knowledge, there are no methods for the Web document similarity retrieval problem (WDSRP) based on randomized query generation and meta-search scoring functions, using mainly the search engines' ranking. The described Zipf-like scoring function can be used as a relevance estimator for a Web search engine result. Also, our probabilistic language model for query generation allows to extract relevant terms, where its weighting approach parameters are sufficient for a key term extraction light-technique.

Acknowledgment

Authors would like to thank continuous support of "Instituto Sistemas Complejos de Ingeniería" (ICM: P-05-004- F, CONICYT: FBO16); FONDEF project (DO8I-1015) entitled, DOCODE: Document Copy Detection (`www.docode.cl`); and the Web Intelligence Research Group (`wi.dii.uchile.cl`).

References

1. Baeza-Yates, R.A., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston (1999)
2. Hakerness, W.L.: Properties of the extended hypergeometric distribution. Ann. Math. Statist. 36(3), 938–945 (1965)
3. Henzinger, M.: Finding near-duplicate web pages: a large-scale evaluation of algorithms. In: SIGIR 2006: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 284–291. ACM, New York (2006)
4. Pereira Jr., A.R., Ziviani, N.: Retrieving similar documents from the web. J. Web Eng. 2(4), 247–261 (2004)
5. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
6. Nagaraj, S.V.: Web Caching And Its Applications. Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, Norwell (2004)
7. Selberg, E., Etzioni, O.: The metacrawler architecture for resource aggregation on the web. IEEE Expert, 11–14 (January–February 1997)
8. Somlo, G.L., Howe, A.E.: Using web helper agent profiles in query generation. In: AAMAS 2003: Proceedings of the second international joint conference on Autonomous agents and multiagent systems, pp. 812–818. ACM, New York (2003)
9. Zaka, B.: Empowering plagiarism detection with a web services enabled collaborative network. Journal of Information Science and Engineering 25(5), 1391–1403 (2009)
10. Zipf, G.K.: Human Behavior and the Principle of Least Effort. Addison-Wesley, Reading (1949)