Language Model and Clustering based Information Retrieval

Irene Giakoumi, Christos Makris and Yiannis Plegas
Department of Computer Engineering and Informatics, University of Patras, Patra, Greece

Keywords: Clustering, Information Retrieval, Kullback-Leibler Divergence, Language Models, Query Expansion, Redundant Elimination, Semantics.

Abstract: In this paper, we describe two novel frameworks for improving search results. Both of them organize relevant documents into clusters utilizing a new soft clustering method and language models. The first framework is query-independent and takes into account only the inter-document lexical or semantic similarities in order to form clusters. Also, we try to locate the duplicated content inside the formed clusters. The second framework is query-dependent and uses a query expansion technique for the cluster formation. The experimental evaluation demonstrates that the proposed method performs well in the majority of the results.

1 INTRODUCTION

Information retrieval aims at satisfying an information need by ranking documents optimally. In retrieval systems, an information need is expressed in the form of a query (Manning et al., 2008). The goal is to rank relevant documents higher than the non-relevant. Clearly, the performance of an information retrieval system is determined by the chosen retrieval function. A retrieval model formalizes the notion of relevance and derives a retrieval function that can be computed to rank documents.

Effective retrieval functions have been derived from a class of probabilistic models, the language modelling approaches. Essentially, the idea is the computation of probabilistic distributions over documents or collections of documents. Several approaches have proven the effectiveness of this method for information retrieval.

Recently, language models have been used in combination with another approach of information retrieval, which is the cluster-based retrieval. Under this line, documents relevant to a query are grouped into clusters and the rank of them depends on the cluster that they belong to. Cluster-based retrieval is based on the cluster hypothesis according to which similar documents will satisfy the same query (Rijsbergen, 1979). Its aim is to identify the good clusters for the given query. Recent researches have shown that if these good clusters were able to be found then retrieval performance would be improved over document-based retrieval, where the query is matched against documents (Raiber and Kurland, 2012).

An issue that affects the performance of information retrieval systems is the content duplication, a problem known as redundant elimination. The same information is often met in more than one documents, while the ideal answer to a query should be all the unique information in descending order of relevance. The solution of this problem remains a challenge.

In this paper, we propose two new frameworks in order to improve query search results by forming and selecting the good clusters cluster-based retrieval searches for. Both of the frameworks form clusters of relevant documents using a new soft clustering method and the language modelling approach. The first proposed method examines, query-independently, either lexical or semantic inter-document similarities in order to form clusters. In order to further improve search results, we examine the case of locating redundant information in the clusters of relevant documents our method forms. To do so, we implement and apply an existing redundant elimination method over clusters. The second one creates a cluster for every possible query expansion, when semantics of query are taken into account. Our final results are lists of documents with improved ranking.

The rest of this paper is organized as follows: We briefly survey language models, cluster-based retrieval and the redundant elimination problem in Section 2, while in Section 3 we describe the language models estimation. We detail our frameworks for improving search results in a query independent way in Section 4 and in a query dependent way, using a query

Giakoumi I., Makris C. and Plegas Y..
Language Model and Clustering based Information Retrieval.
DOI: 10.5220/0005409804790486
In *Proceedings of the 11th International Conference on Web Information Systems and Technologies* (WEBIST-2015), pages 479-486
ISBN: 978-989-758-106-9

expansion technique in Section 5. Section 6 contains the experimental evaluation of our methods and we conclude in Section 7.

2 RELATED WORK

A statistical language model assigns probabilities to a sequence of words by means of a probability distribution. This concept has been used in various applications such as speech recognition and machine translation. The first to employ language models for information retrieval were Ponte and Croft (Ponte and Croft, 1998). Their basic idea was: estimate a language model for each document and rank documents by the likelihood scoring method. Since then several variants of this basic method have been proposed as well as several improvements of it.

Zhai and Lafferty (Zhai and Lafferty, 2004) have proved the importance of the selection of smoothing parameters. The term smoothing refers to the adjustment of the maximum likelihood estimator of a language model so that it will be more accurate and at least it will not assign zero probability to words that are not met in a document, which is known as the zero probability problem.

Recent researches attempted to exploit the corpus structure, which is, clusters of documents used as a form of document smoothing using the language modelling retrieval framework. Language models over topics are constructed from clusters and the documents are smoothed with these topic models to improve document retrieval ((Kurland and Lee, 2004); (Liu, 2006); (Liu and Croft, 2004)). In this line Liu and Croft (Liu and Croft, 2004) cluster documents and smooth a document with the cluster containing the document. Also, Kurland and Lee (Kurland and Lee, 2004) suggested a framework which for each document obtains the most similar documents in the collection and then smooths the document with the obtained ”neighbour documents”. These neighbourhoods of documents are formatted using as measure the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951) which is an asymmetric measure of how different two probability distributions are. This measure has been used in several researches ((Lafferty and Zhai, 2001); (Kurland, 2006)).

The use of clusters for smoothing purposes is an approach to cluster-based retrieval, while the most common approach aims to create and retrieve one or more clusters in response to a query. Cluster-based retrieval is an idea that has a long history and many researchers have worked on it. Using as base the cluster hypothesis (Rijsbergen, 1979) several hard clustering methods were employed ((Croft, 1980); (Jardine and van Rijsbergen, 1971); (Voorhees, 1985)) as well as several probabilistic (soft) clustering methods ((Blei et al., 2003); (Hofmann, 2001)). The soft clustering methods accept the following case: a document could discuss multiple topics, and if one assumes that clusters represent topics, then the document should be associated with the corresponding clusters. This is the case that is accepted in this paper.

Finally, another important problem in information retrieval, that is needed to refer to, is the problem of information redundancy. Several approaches have been proposed attempting to resolve this problem. Some of them use ranking algorithms in order to reduce the redundant information from the search results ((Agrawal et al., 2009); (Chen and Karger, 2006); (Radlinski et al., 2009)), while some others reward novelty and diversity covering all aspects of a topic ((Agrawal et al., 2009); (Clarke et al., 2008)). An interesting approach (Plegas and Stamou, 2013) extracts the novel information between documents with semantically equivalent content and creates a single text, called SuperText, relieved from the duplicated content. The suggestion we make in this paper is that clusters of documents, assuming that clusters represent topics, are an ideal group of documents to locate repeated information.

Since a query could have many interpretations and the documents retrieved for a query could discuss different aspects of the same topic, the ranking of documents in results list should take these parameters into account. Being inspired from the related work previously referred, we consider the idea of forming overlapping clusters of relevant documents, taking advantage of the asymmetry of KL-divergence when it is applied over document language models. The clustering method, we suggest, not only locates pairs of similar documents, but also locates which of the two is more similar to the other. The first method we suggest locates inter-document lexical similarities using our soft clustering method applied over document language models. Subsequently, we implement the same idea, but this time we try to locate inter-document semantic similarities. In order to do this, we examine a new way to embed semantic information within document language models. This method is query-independent, since the query is being used only for ranking purposes. As we previously referred, we consider the case of locating duplicated content inside the formed clusters, to further improve search results. For this purpose, we apply a redundant elimination method (Plegas and Stamou, 2013) not over the initial search results list, but over each of our clusters, which is also a new try to remove duplicated informa-

tion. On the other hand, the second method we propose is query dependent as the query is important for the cluster formation. Specifically, we apply an existing query expansion method (Makris et al., 2012) to form all possible interpretations of a query and using our clustering method we form clusters that answer to every expansion of the query. This method takes into account both lexical and semantic information of the query in order to calculate the document language models. The novelty of this method is located into the way that the expansions of a query are utilized. In the following sections we describe in detail the suggested frameworks.

3 LANGUAGE MODEL ESTIMATION

We represent the information contained in the documents of query results with the language modelling approach. The basic idea of this method is the calculation of a probability measure over strings that belong to a fixed vocabulary V (Manning et al., 2008). That is, considering a fixed set of strings, a document language model calculates the probability of locating these strings in the document. In our case, vocabulary V consists of words or senses. For simplicity, we chose to use the Smoothed Maximum Likelihood Estimate (SMLE) model in our approach. According to the Maximum Likelihood Estimate (MLE) model the representation of a document d is based on the number of occurrences of each term t in a fixed vocabulary V:

$$M_d^{MLE}(t) = \frac{f_{t,d}}{l_d}, \forall t \in V \quad (1)$$

,where $f_{t,d}$ is the number of occurrences of term t in document d and l_d is the number of terms contained in d and also in V. The MLE model suffers from the zero probability problem, that is:

$$M_d^{MLE}(t) = 0, \forall t \in V \,\&\, \forall t \ni d \quad (2)$$

In order to solve it, we smooth the model according to the following formula:

$$M_d^{SMLE}(t) = \begin{cases} \frac{f_{t,d}}{l_d} - c, \text{if } t \in d \\ \text{eps}, \text{if } t \ni d \end{cases}, \forall t \in V \quad (3)$$

,where eps is a very small quantity of the order of 10^{-10} and c is estimated as

$$c = eps * \frac{|V| - s_d}{|V|} \quad (4)$$

,where s_d is the number of terms that belong in V and the document d and $|V|$ is the length of vocabulary used.

4 CLUSTERING WITH LEXICAL OR SEMANTIC LANGUAGE MODELS

In this section we describe our method for locating inter-document similarities and forming clusters utilizing the Kullback-Leibler divergence over lexically or semantically enhanced language models. Here the issued query is taken into account only for ranking purposes, not for the creation of the clusters.

4.1 Document Representation

In our experiments, we use lexically or semantically enhanced language models. The difference between their estimation concerns the fixed vocabulary V. But before describing the set V in each case, we need to describe the initial processing of the search result list.

We select the top N documents retrieved for a query, which will be called collection D. We break each document d in collection D into sentences. Each sentence is tokenized, POS-tagged and lemmatized and all lemmas are removed but the ones that correspond to nouns, verbs and adjectives. The set of processed documents of collection D will be denoted as collection D'. In a similar way we process the issued query. Now that collection's D' documents and the query contain only certain lemmatized terms, along with their POS-tag, we describe the fixed vocabulary V for our language models.

The set V of the lexically enhanced language models consists of all the unique POS-tagged lemmas contained in the collection D'. In other words, we examine the inter-document similarities considering only the vocabulary that the documents contain. For reasons of dimensionality reduction, we evaluate also the case of reducing the size of V, using the normalized TF-IDF schema. In this case, the terms in V are ordered by their TF-IDF values and the desired amount of terms in V is selected. Also the content of documents in D' is updated, so that they contain only terms of V.

In the case of semantically enhanced language models, firstly, all the terms contained in collection D' are issued to WordNet, in order to locate all the possible senses each term can have. Continuing, we map all terms of a document to their corresponding sense. Words matching only one sense are annotated with it. Words matching several senses are annotated with the sense that exhibits the maximum average similarity to the senses identified for the remaining document words. We estimate semantic similarity based on the Wu and Palmer metric (Wu and Palmer, 1994). The

similarity between two senses s_i and s_j from two terms w_i and w_j is given by:

$$\text{Similarity}(s_i, s_j) = \frac{2 * \text{depth}(\text{LCS}(s_i, s_j))}{\text{depth}(s_i) + \text{depth}(s_j)} \quad (5)$$

Since the appropriate senses for w_i and w_j are not known, our measure selects the senses which maximize *Similarity* in order to annotate every term in the document with an appropriate sense. Here, the similarity between documents is examined over the senses each one contains. The set of the unique senses annotated to the documents of D' will be used as the set V, considering, also, the option of reducing its size, in a way similar to the lexical language model case using the TF-IDF schema.

After this procedure, which is necessary for the language model estimation, we have the needed fixed vocabulary V and the contents of documents in D' in the appropriate form. Using the formula described in equation (3) we compute the lexically or semantically enhanced language models for the N documents.

4.2 Soft Clustering Method

Now that documents are represented by probability distributions, we focus on locating the similar ones using KL divergence. Its value is given by:

$$\text{KLD}(f(x) \| g(x)) = \sum_x f(x) * \log \frac{f(x)}{g(x)} \quad (6)$$

,where f(x) and g(x) are discrete probability distributions. The smaller the KL-divergence's values are, the more similar are the distributions. The smallest valid value is zero indicating that the distributions are identical. KL-divergence is non-symmetric, a characteristic that we take advantage of, in contrast to other research (Liu and Croft, 2004). So, it could be considered that formula (6) measures the information loss if distribution g(x) replaces f(x), while $\text{KLD}(g(x) \| f(x))$ measures the information loss if distribution f(x) replaces g(x). In our case the probability distributions represent texts. That is, calculating the values of KL-divergence between documents we i) locate documents that have similar vocabulary or senses and so documents referring to the same topic and ii) examine between two documents which is more similar to the other, that is which contains less new information. Taking advantage of the asymmetry of KL-divergence, we estimate:

$$\begin{aligned} &\text{KLD}(M_{d_i}^{SMLE}(t) \| M_{d_j}^{SMLE}(t)) \,\& \\ &\text{KLD}(M_{d_j}^{SMLE}(t) \| M_{d_i}^{SMLE}(t), \\ &\quad \forall d_i, d_j \in D' \,\&\, i \neq j. \end{aligned} \quad (7)$$

Initially, we consider that each document of D' forms a singleton cluster. For every pair of documents we compare their both estimated KL-divergence values and if at least one of them is small enough, then the two documents are lexically or semantically (depending on which set V is used) similar and should belong to the same cluster. We determine a parameter t as an upper bound for the values of KL-divergence of similar documents. The decision of which document will be considered similar to another is taken according to the following formula:

$$\begin{aligned} &\text{if}\{\text{KLD}(d_i \| d_j) \leq t \text{ OR } \text{KLD}(d_j \| d_i) \leq t\} \\ &\text{then} \begin{cases} d_i \in \text{Cl}(d_j), & \text{iff } \text{KLD}(d_i \| d_j) < \text{KLD}(d_j \| d_i) \\ d_j \in \text{Cl}(d_i), & \text{iff } \text{KLD}(d_j \| d_i) < \text{KLD}(d_i \| d_j) \end{cases} \\ &\forall i, j \text{where } i \neq j \& 1 < i, j < |D|. \end{aligned} \quad (8)$$

,where $\text{Cl}(d_i)$ is the cluster which initially contained document d_i. For every cluster we consider as basis the document that it initially contained. During the process of clustering some of the resulting clusters may appear as members of a bigger cluster. In this case, the internal clusters are eliminated and only the external is kept without removing any of its containing documents.

4.3 Removing Redundant Information

At this point, we have organized the documents into overlapping clusters of similar documents. We claim that only documents in the same cluster have duplicated information. In order to locate and remove it, we merge the documents of each cluster into a single text keeping all the unique information only, an idea that origins from (Plegas and Stamou, 2013). If, after clustering, a cluster contains only its basis document, then that document remains as it is. If a cluster contains more than one document, then to the content of the basis document is added the new unique information contained in the other documents in the cluster (member documents).

4.4 Ranking

The process described in the previous section results in a new collection of documents, which contains texts generated from more than one documents contained in the same cluster and possibly some of the documents initially retrieved for the query (clusters that remained singleton). The last step of our method is to present this collection in a ranked list of decreasing order of interest. We estimate the lexical enhanced language model for all the documents in the fi-

nal collection and for the query, using as fixed vocabulary the POS-tagged lemmas contained in the query. At this point we choose not to use senses, but only terms, for the estimation of language models, because firstly the usually small number of terms contained in a query does not help the application of a word sense disambiguation method and secondly we would not want to focus on a certain interpretation of the query. Having estimated the language models, we estimate how similar the language model of each document is to the language model of the query by the corresponding values of KL-divergence. The ranking of the values of KL-divergence in increasing order derives the desired ranking list of the documents.

5 CLUSTERING USING QUERY EXPANSION

This section describes our method for the clusters' creation, taking into account the different interpretations a query could have. We attempt to form clusters of documents, that each answers to a specific expansion of the query, that is clusters that answer to a specific interpretation of it.

5.1 Document Representation

In order to represent the contents of the documents in results' list we use, this time, only lexical language models. We made this choice because the language models are calculated relying on the query expansions as will be explained in this and the following section.

Again from the results' list we use the top N documents, this is collection D. As in the previous method, we break the documents into sentences and each sentence into tokens, which subsequently are POS-tagged, lemmatized and reduced in size as we keep lemmas of nouns, verbs and adjectives. Again, the processed collection D we call it collection D'.

5.2 Expanding the Query

Query expansion is the process of reformulating the issued query to improve retrieval performance (Baeza-Yates and Ribeiro-Neto, 2011). In our method, we expand the queries employing an idea based on a already existing method. The idea is to replace the initial query with a set of queries (collection Q), each one containing a different combination of senses of the query terms (Makris et al., 2012).

The query expansion procedure operates as follows: Firstly we tokenize and label with POS-tags the query terms, and we keep only lemmas of nouns, verbs and adjectives as we did for the collection D. The next step is to issue each of these terms into WordNet. This results in a set of senses for every query term. For every sense WordNet gives a definition for it, called gloss. We process these glosses in order to extract from them lemmas of nouns, verbs and adjectives with the same procedure described previously. The final step to create the collection Q of expanded queries is to consider all possible combinations of senses for every term and add the extracted terms from glosses to the initial query. Before clustering, we need to compute language models over the documents in collection D' and for that we need a fixed vocabulary V. We want to form a cluster for each expanded query, so we choose to use as fixed vocabulary the terms contained in each expanded query. The language models of documents are estimated as described in Section 3.

5.3 Clustering

After we have computed the language model of each expanded query and of the documents as described in the previous subsection, we use a clustering method similar to the one described in subsection 4.2.

$$\text{if } \mathrm{KLD}(q_i \| d_j) \leq t \text{ then } d_j \in \mathrm{Cl}(q_i), \quad \forall q_i \in Q \;\&\; \forall d_j \in D'. \tag{9}$$

Here, the Kullback-Leibler divergence values are calculated over the language model of every expanded query and every document in order to evaluate which probability distributions of the documents are similar to the expanded query's distribution. The result of this process is a cluster for every expansion of the query. These clusters may be overlapping. Since the length of the fixed vocabulary V is generally quite short, we evaluate our clustering method for different values of the parameter t.

5.4 Ranking

At this stage we have as many clusters as the expanded queries. The last step is to rank our results. First we rank the documents in each cluster according to the Kullback-Leibler divergence values calculated previously. The documents are sorted in ascending order. Finally, we rank the clusters. In order to reduce the computational cost we rank the clusters and consequently the expanded queries, according to the number of documents each cluster has. The intuition behind this choice is that we choose to place at the top positions the expanded query for which most of the documents in collection D have information.

6 EXPERIMENTAL EVALUATION

To carry out our evaluation, we explored the same 70 web queries for both our of methods from the TREC WebTracks(2011, 2012). We selected the queries with the best characteristics in order to perform our experiments. All tracks use the 1 billion page ClueWeb09 (http://lemurproject.org/clueweb09/) dataset and contain a diversity task that contains a ranked list of documents that covers the query topic avoiding information redundancy. For every result there is a relevance judgement by NIST assessors indicating their relevance with their query topic. Also we have employed the search engine Indri (http://www.lemurproject.org/indri/) and for each query we retained the top 50 results for both evaluating methods. In all of the figures that follow, vertical axis represents the a-nDCG values, while horizontal axis represents the rank position of documents in final results' list.

6.1 Evaluating the First Method

We created clusters of similar documents and the upper bound of KL-Divergence value, parameter t, was experimentally tested and took the value of t=10.0. We evaluated our results keeping 20%, 40%, 60%, 80% and 100% of the terms/senses in the fixed vocabulary V. We compared our techniques: semantic (semantically enhanced) and lexical (lexical enhanced) with an approach that used k-means as clustering algorithm (we did that to test the effect of the clustering method).

Finally we assessed the performance of the proposed methods by comparing their rankings with the initial rankings of search engine Indri. We measured the efficiency of the proposed methods using (i) the relevance judgements and (ii) the a-nDCG (a -normalized Discounted Cumulative Gain) measure (Clarke et al., 2008) with a=0 where the value of 1.0 is a good indicator.

Figure 1 contains the results for our method when we kept the 60% of the terms/senses, as this is the best case. The results of the other four examined cases are very close to the presented one.

According to the results our method exceeds baseline. Also k-means is better than our clustering method. This is an indicator that a less "strict" value of t is needed. Results show, also, that semantically enhanced language models do not give as good results as lexical language models do at the top rank positions. We claim that, this indicates that semantically enhanced language models are more sensitive when calculating them.

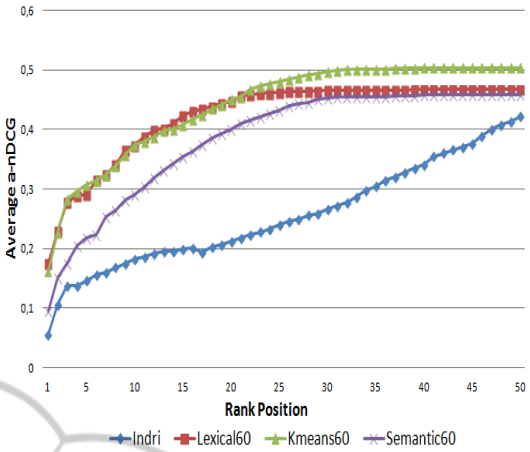

Figure 1: Average a-NDCG value for every rank position keeping the 60% of terms/senses.

6.2 Evaluating the Second Method

With this method we created clusters for every expansion of the queries. Because of the short length of the fixed vocabulary used, the value of t=10.0 could be considered quite "strict". For this reason we evaluated our method for the following values of parameter t: 10.0, 12.0, 14.0, 16.0, 18.0 and 20.0.

Figure 2 contains the results of our method. This method, also, exceeds baseline and shows that value 16.0 gives the best results, which confirms our argument about the need of a less strict and not very tolerant value for parameter t. We should also note that this method is suitable for queries that their terms are contained into WordNet.

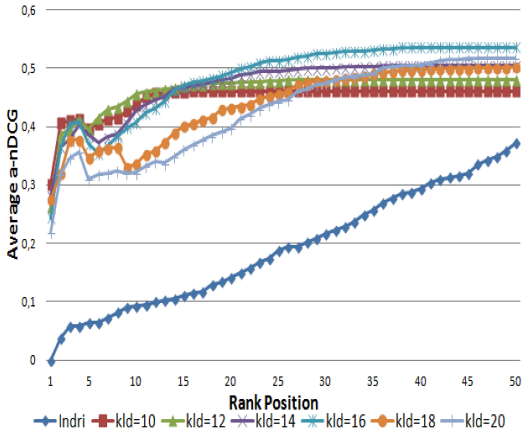

Figure 2: Average a-nDCG value for every rank position.

Finally, we compared the best results of the two proposed methods. Figure 3, shows that clustering in a query-dependent way gives a better ranking compared with the case of clustering independently to

the query. This could be considered justified, since clusters' formation based on query's interpretations is more adjustable and less sensitive to the choices of the documents' authors. Also, our clustering algorithm gives better results than k-means when applied to the second proposed method.

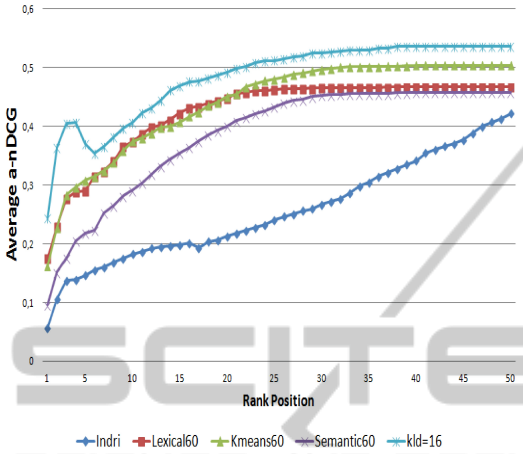

Figure 3: Average a-nDCG value for every rank position when comparing the two proposed methods.

Previous results show that in all cases, our proposed methods perform better than the baseline of Indri search engine. That is encouraging and shows that our methods have many good results to demonstrate. In order to improve our results our next step is to improve the performance of the semantically enhanced language models, as we claim that is a valuable tool and could even exceed the performance of lexical language models. Also, we intend to do further research on the performance of our clustering method and compare it with more clustering algorithms. We believe that our methods could give important results when comparing our results with the application of different language models, smoothing methods and redundant elimination methods.

7 CONCLUSIONS

In this paper we presented two novel frameworks that improve the results' list of a query. The first method creates overlapping clusters using our proposed clustering method and lexically or semantically enhanced language models. Additionally, it removes the redundant content from the clusters. The second method, we propose, creates a set of overlapping clusters, where each cluster answers to one of the interpretations of the initial query. The novelty of these frameworks is the suggested clustering method, that utilizes the asymmetry of KL-Divergence, the way of semantic enhance of language models, the use of a query expansion technique in order to form clusters, that each answers to an interpretation of the initial query and also the application of a redundant elimination method over clusters. The experimental results indicated that the proposed methods perform quite well in relation to the state of the art techniques.

ACKNOWLEDGEMENTS

This research has been co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Thales. Investing in knowledge society through the European Social Fund

REFERENCES

Agrawal, R., Gollapudi, S., Halverson, A., and Ieong, S. (2009). Diversifying search results. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*. ACM.

Baeza-Yates, R. and Ribeiro-Neto, B. (2011). *Modern Information Retrieval: the concepts and technology behind search*. ACM Press, 2nd edition.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Chen, H. and Karger, D. R. (2006). Less is more: Probabilistic models for retrieving fewer relevant documents. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 429–436. ACM.

Clarke, C., Kolla, M., Cormack, G., Vechtomova, O., Ashkan, A., Buttcher, S., and MacKinnon, I. (2008). Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 659–666. ACM.

Croft, W. (1980). A model of cluster searching based on classification. *Information Systems*, 5(3):189 – 195.

Hofmann, T. (2001). Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1–2):177–196.

Jardine, N. and van Rijsbergen, C. (1971). The use of hierarchic clustering in information retrieval. *Information Storage and Retrieval*, 7(5):217 – 240.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *Ann. Math. Statist.*, 22(1):79–86.

Kurland, O. (2006). *Inter-document Similarities, Language Models, and Ad Hoc Information Retrieval*. PhD thesis, Cornell University.

Kurland, O. and Lee, L. (2004). Corpus structure, language models, and ad hoc information retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 194–201. ACM.

Lafferty, J. and Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 111–119. ACM.

Liu, X. (2006). Cluster-based retrieval from a language-modeling perspective. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 737–738.

Liu, X. and Croft, W. B. (2004). Cluster-based retrieval using language models. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 186–193. ACM.

Makris, C., Plegas, Y., and Stamou, S. (2012). Web query disambiguation using pagerank. *JASIST*, 63:1581–1592.

Manning, C. D., Raghavan, P., and Schutze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Plegas, Y. and Stamou, S. (2013). Reducing information redundancy in search results. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, SAC '13, pages 886–893. ACM.

Ponte, J. M. and Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 275–281. ACM.

Radlinski, F., Bennett, P. N., Carterette, B., and Joachims, T. (2009). Redundancy, diversity and interdependent document relevance. *SIGIR Forum*, 43(2).

Raiber, F. and Kurland, O. (2012). Exploring the cluster hypothesis, and cluster-based retrieval, over the web. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 2507–2510. ACM.

Rijsbergen, C. J. V. (1979). *Information Retrieval*. Butterworth-Heinemann, 2nd edition.

Voorhees, E. M. (1985). The cluster hypothesis revisited. In *Proceedings of the 8th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '85, pages 188–196. ACM.

Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics*, ACL '94, pages 133–138. Association for Computational Linguistics.

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