Integrating Semantic Term Relations into Information Retrieval Systems Based on Language Models

Mohannad ALMasri[1], KianLam Tan[1], Catherine Berrut[1], Jean-Pierre Chevallet[2], and Philippe Mulhem[3]

[1] Université Joseph Fourier - Grenoble 1
{mohannad.almasri,kian-lam.tan,catherine.berrut}@imag.fr
[2] Université Pierre Mendès - Grenoble 2
jean-pierre.chevallet@imag.fr
[3] Centre National de la Recherche Scientifique
philippe.mulhem@imag.fr
LIG laboratory, MRIM group, Grenoble, France

Abstract. Most information retrieval systems rely on the strict equality of terms between document and query in order to retrieve relevant documents to a given query. The term mismatch problem appears when users and documents' authors use different terms to express the same meaning. Statistical translation models are proposed as an effective way to adapt language models in order to mitigate term mismatch problem by exploiting semantic relations between terms. However, translation probability estimation is shown as a crucial and a hard practice within statistical translation models. Therefore, we present an alternative approach to statistical translation models that formally incorporates semantic relations between indexing terms into language models. Experiments on different CLEF corpora from the medical domain show a statistically significant improvement over the ordinary language models, and mostly better than translation models in retrieval performance. The improvement is related to the rate of general terms and their distribution inside the queries.

1 Introduction

Classical retrieval models are primarily based on exact matching of terms between documents and queries, and are unable to capture relations between queries and documents terms. When users utilize different terms from those that are used in the index to express the same meaning, classical retrieval models suffer from term mismatch problem. Consequently, in order to overcome the term mismatch problem, we need a retrieval model that captures semantic relations between terms. Term relations are normally obtained from an external knowledge or resource[1]. The integration of term relations contributes to reduce the gap during the matching between a query representation and a document

[1] External knowledge with respect to queries and documents like thesaurus or ontology.

index. Several types of semantic relations are identified between terms. We only focus on hierarchical relations or specific-generic relations.

Term specificity is a semantic property that can be applied to index terms: a term is more specific when its meaning is more detailed and precise. Term specificity may cause a mismatch when a user formulates her/his query using terms which are more general than those in the index. For instance, in the medical domain, the terms "*B-Cell*" and "*T-Cell*" are more specific than the term "*Lymphocyte*". Moreover, "*B-Cell*" and "*T-Cell*" are types of "*Lymphocyte*" in the adaptive immune system. Therefore, when a user query contains the term "*Lymphocyte*", then, a document talking about "*B-Cell*" or "*T-Cell*" is relevant to this query.

Term mismatch has been heavily studied in IR, and various approaches have been proposed to overcome it, generally through a pragmatic or an ad-hoc approach. However, a few of them have focused on formal integration of semantic term relations into retrieval models. Inside the family of language models for IR, the statistical translation models have been shown as an effective way to mitigate the term mismatch problem [3] [9] [20]. Statistical Translation models integrate semantic relations into language models to reduce the gap between documents and queries, but they require an estimation of the translation probabilities between words which is a crucial and hard point. We propose in this paper an alternative approach rather than statistical translation models to formally integrate semantic term relations into the framework of language models. Our approach propose to modify a document according to a given query and some knowledge about term relations. We then integrate the modified document into two smoothing methods from language models: Dirichlet and Jelinek-Mercer. In the rest of the paper, we refer by a term to an indexing term, which can be either: a word, a noun phrase, a n-gram, or a concept [5].

The paper is organized as follows: first, we present the term mismatch problem, and we discuss several approaches to solve this problem in Section 2 followed by our approach presented in Section 3. Our experimental set-up and the empirical results are presented in section 4; finally, section 5 concludes the paper and presents the future work.

2 Term Mismatch Problem

Several techniques have been proposed to tackle the term mismatch problem. Among these techniques: relevance feedback [11] [17], dimension reduction [2] [7] [8] [10] [16], and integrating term similarity into retrieval models [3] [6] [9].

2.1 Relevance Feedback

Relevance feedback involves the user in the IR process in order to reformulate her/his query and to improve retrieval results. There are three types of relevance feedback: 1) explicit feedback, 2) implicit feedback and 3) pseudo or blind feedback [13]. Rocchio algorithm [17] is the classic algorithm for implementing

explicit feedback which enables the user to select relevant documents in order to reformulate the original query. Query Reformulation is achieved by adding terms extracted from the selected documents into the original query.

Implicit feedback incorporates user behavior like clicks or the duration of time spent viewing a document, in order to predict relevant documents which are used to reformulate a user query. Blind feedback automates the manual part of the Rocchio algorithm without any consideration of the user interaction by assuming that the top k ranked documents are relevant. Lavrenko and Croft [11] propose a blind feedback approach to estimate a relevance model. The main problem in implicit and explicit feedback is that they may cause query drift[2] because not all documents in the feedback set may be relevant. Besides, documents in the feedback set, although containing relevant information, are sometimes partially related to the query topic.

2.2 Dimension Reduction

Dimension reduction is the process of reducing the chance that a query and a document use different terms for representing the same meaning. Among the techniques that are used for achieving this mission, we can mention: Stemming [10] [14] [16], Latent Semantic Indexing (LSI) [7], and Conceptual Indexing [2] [5] [12]. These techniques propose different strategies to reduce the chances that the query and the document refer to the same concept but using different terms.

Peng et al. [14] perform a stemming method according to the context of the query which helps to improve the accuracy and the performance of retrieval compared to the query independent stemmers such as Porter [16] and Krovetz [10]. Deerwester et al. [7] propose to solve the dimension reduction by representing queries and documents in a latent semantic space. In latent semantic space, each term is grouped with its similar terms. Similar terms in the space tend to be the terms that share the same context. The context is: a sentence, a paragraph, a window of successive words, etc..

Effectiveness of dimension reduction techniques essentially depend on the application domain and on the characteristics of the studied collection. Besides, dimension reduction may cause an oversimplification of the term space that may limit the expressiveness of the indexing language and could result in incorrect classification of unrelated terms [6].

2.3 Exploiting Term Similarity

We present, in this section, a class of retrieval models that attempt to solve the term mismatch problem by exploiting a partial or complete knowledge of term similarity. The use of term similarity enables to enhance classical retrieval by taking into account non-matching terms. We present two categories of models: Vector Space and Language Models.

[2] Query drift is the change in the query's topic to an unintended direction after query expansion.

Term Similarity in Vector Space Models Crestani [6] proposes a general framework to exploit the term similarity into the matching process where $w_d(t)$ is the weight assigned to term t in the document d, and $w_q(t)$ is the weight assigned to term t in the query q, as shown below:

$$RSV(d,q) = \sum_{t \in q} w_d(t) \times w_q(t) \tag{1}$$

In order to consider the non-matching terms from the query, Crestani exploits the term similarity by utilizing a similarity function Sim. If $t_i = t_j$, then $Sim(t_i, t_j) = 1$. If t_i and t_j are semantically related, then $0 < Sim(t_i, t_j) < 1$ and otherwise it is 0.

In fact, Crestani proposes to extend the previous RSV of Eq.1 in two ways. First, he extends the matching process, in case of mismatch: $t \in q$ and $t \notin d$ by determining the closest document term t^*, for which we have maximum value of similarity with the query term t. As a result, the extended RSV_{max} is defined:

$$RSV_{max}(d,q) = \sum_{t \in q} Sim(t, t^*) \times w_d(t^*) \times w_q(t) \tag{2}$$

when $t = t^*$ then $Sim(t, t^*) = 1$, and we return back to the Eq.1.

Second, Crestani also extends RSV of Eq.1 by considering, not only the most similar term, but all the related terms from the document to a non-matched query term. As a result, the extended RSV_{tot} is:

$$RSV_{tot}(d,q) = \sum_{t \in q} \big[\sum_{t' \in d} Sim(t, t') \times w_d(t') \big] \times w_q(t) \tag{3}$$

Crestani integrates term similarity into vector space model, which is an outdated model in information retrieval. We propose to integrate term similarity into language models which have been proven as very effective method for text retrieval [15] [21].

Term Similarity in Language Models Statistical translation models are shown as an effective way to mitigate the term mismatch [3] [9] [20]. Statistical translation models incorporate semantic relations between terms into language models to reduce the gap between documents and queries. The idea is based on information theory where a translation model estimates the probability of translating a document to a user query according to the probability distribution $P(u|v)$, which gives the probability of translating a word v into a word u.

Statistical translation models [3] [9] are related to the second proposition of Crestani [6] where the idea is to consider the similarity between each query term and all document terms. The results obtained by statistical translation models show that integrating term similarity into language models is more effective than the existing approaches in information retrieval. However, Karimzadehgan and Zhai [9] noticed that the self-translation probabilities lead to non-optimal retrieval performance because it is possible that the value of $P(w|u)$ is higher

than $P(w|w)$ for a word, w. In order to overcome this problem, Karimzadehgan and Zhai [9] defined a parameter to control the effect of the self-translation.

In a nutshell, we can remark that statistical translation models represent similarity between terms as a translation probability which may cause some problems: 1) the estimation of translation probabilities is not an easy practice, 2) the normalization of the mutual information is rather artificial and requires a parameter to control the effect of the self-translation, and 3) the regularization of the translation probabilities may look uncertain. Therefore, in the next section, we present an alternative approach that is based on a similarity measure between terms rather than translation probabilities. We believe that our approach is simpler and more efficient than statistical translation models.

3 Integrating Term Relations into Language Models

Referring to the different approaches which are presented in section 2, each approach has its strategy to reduce the potential gap that exists between queries and documents: Relevance feedback modifies the user query by adding some terms in order to shift it toward relevant documents. Dimension reduction represents documents and queries in a new space where the gap between queries and their relevant documents becomes smaller. Exploiting term similarity approaches integrate semantic relations between terms into retrieval models in order to reduce the gap between documents and queries.

We present, in this section, our approach that integrates semantic term relations into language models. Our approach modifies documents according to a given query and some knowledge about term relations. We then estimate the document language model according to the modified document in two smoothing methods of language models: Dirichlet and Jelinek-Mercer.

3.1 Query and Knowledge Dependent Document Model

Our aim is to reduce the gap between documents and queries by considering semantic relations between query and document terms. To do this, we propose to modify a document index according to the query and the external knowledge about term relations. Classical IR models compute the relevance value between a document d and a query q based on the coordination level, namely $d \cap q$. Instead of that, we here propose to compute the relevance value by considering also the *unmatched terms* of the query $t \in q \setminus d$, where $\setminus$ is the set difference operator. We therefore expand d by the query terms that are not in the document, but they are semantically related to at least one document term. In this way, we maximize the coordination level between the modified document and the query. As a result, we maximize the probability of retrieving relevant documents for a given query. We follow the first idea of Crestani presented in Eq.2.

Figure 1 illustrates how we expand d using the external knowledge, denoted by K, in order to maximize the coordination level between d and q. To put it

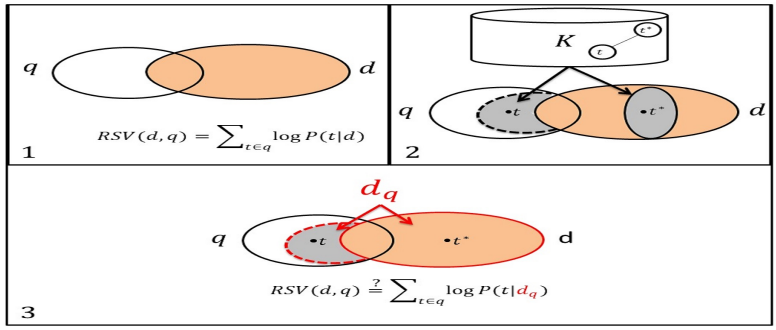

Fig. 1. Expand the document d using the knowledge K. We see that we expand d in order to maximize the coordination level between d and q.

more formally, the modified document, denoted by d_q, is calculated as follows:

$$d_q = d \cup F(q \setminus d, K, d) \tag{4}$$

where $F(q \setminus d, K, d)$ is the transformation of $q \setminus d$ according to the knowledge K and the document d. The knowledge K provides a similarity function between terms $Sim(t, t')$ denoting the strength of the semantic similarity between the two terms t and t', see section 3.5. For each term in the query's unmatched terms $t \in q \setminus d$, we look for a document term t^* which is given by:

$$t^* = argmax_{t' \in d} Sim(t, t') \tag{5}$$

t^* is the most similar term of d for $t \in q \setminus d$. Then, the pseudo occurrences of a query term t in the modified document d_q rely on the occurrences of its most similar document term $\#(t^*; d)$, we define the pseudo occurrences of t as follows:

$$\#(t; d_q) = \#(t^*; d).Sim(t, t^*) \tag{6}$$

this pseudo occurrences of the term t are then included into the modified document d_q. Based on this definition, we now define the transformation function F which expands the document.

$$F(q \setminus d, K, d) = \{t | t \in q \setminus d, \exists t^* \in d, t^* = argmax_{t' \in d} Sim(t, t')\} \tag{7}$$

Not that, if t is not related to any document term, then we do not have a corresponding t^* for t. Then, the unmatched term $t \in q \setminus d$ will not expand d. Now, we replace the the transformation F with its value in the Eq.4 to obtain the modified document as follows:

$$d_q = d \cup \{t | t \in q \setminus d \wedge \exists t^* \in d : t^* = argmax_{t' \in d} Sim(t, t')\} \tag{8}$$

The length of the modified document $|d_q|$ is calculated as follows:

$$|d_q| = |d| + \sum_{t \in q \setminus d} \#(t^*; d).Sim(t, t^*) \tag{9}$$

We see in section 3.3 and 3.4 that the modified document d_q will replace d, and the language models for a query q will be estimated according to the modified document d_q instead of d. We believe that the probability estimation will be more accurate and more effective than ordinary language model.

3.2 Language Models

Language modeling approach to information retrieval is proposed by Ponte and Croft [15]. The basic idea of language models assumes that a query q is generated by a probabilistic model based on a document d. Language models are interested in estimating $P(d|q)$, i.e. the probability that a document d is used to generate query d. By applying Bayes' formula, we have:

$$P(d|q) \propto P(q|d).P(d) \tag{10}$$

$\propto$ means that the two sides give the same ranking. $P(q|d)$ the query likelihood for a given document d. $P(d)$ is often assumed to be uniform and thus can be discarded for ranking documents, then we can rewrite the formula after adding the *log* function as:

$$logP(d|q) = \sum_{t \in V} \#(t;q).logP(t|d) \tag{11}$$

where $\#(t;q)$ is the count of term t in the query q and V is the vocabulary set. Assuming a multinomial distribution, the simplest way to estimate $P(t|d)$ is the maximum likelihood estimator:

$$P_{ml}(t|d) = \frac{\#(t;d)}{|d|} \tag{12}$$

where $|d|$ is the document length. Due to the data spareness problem, the maximum likelihood estimator directly assign *null* to the unseen terms in a document. Smoothing is a technique to assign extra probability mass to the unseen terms in order to solve this problem.

3.3 Extended Dirichlet Smoothing

Dirichlet smoothing [21] is one of the smoothing technique based on adding an extra pseudo term frequency: $\mu P(t|C)$ as follows

$$P_{\mu}(t|d) = \frac{\#(t;d) + \mu P(t|C)}{|d| + \mu} \tag{13}$$

where C is the collection. The main idea of our proposal is to formally integrate term semantic relations into the current Dirichlet formula in order to solve the mismatch. As we mentioned in section 3.1, we assume the case of mismatch: $t \in q$, and $t \notin d$. There is a document term $t^* \in d$ semantically related to t that can play its role during the matching. More specifically, we consider that if t does

not occur in the initial document d, it occurs in the *modified document* d_q, which is the result of expanding d according to the query q and some knowledge[3].

The probability of the term t is defined according to the modified document model d_q. Now, the extended Dirichlet smoothing leads to the following probability for a term $t \in d_q$, $P_\mu(t|d_q)$ which is defined as:

$$P_\mu(t|d_q) = \begin{cases} \frac{\#(t;d)+\mu P(t|C)}{|d_q|+\mu} & \text{if } t \in d \\ \\ \frac{\#(t^*;d).Sim(t,t^*)+\mu P(t|C)}{|d_q|+\mu} & \text{if } t \notin d \end{cases} \tag{14}$$

Note that in the special case where all the query terms occur in the document, we have $|d_q| = |d|$, and that leads to an equal probabilities $p_\mu(t|d) = p_\mu(t|d_q)$.

3.4 Extended Jelinek-Mercer Smoothing

Jelinek-Mercer smoothing [21] is another smoothing technique to add an extra pseudo term frequency: $\lambda P(t|C)$ to the unseen term as follows:

$$P_\lambda(t|d) = (1-\lambda)P(t|d) + \lambda P(t|C) \tag{15}$$

The probability $P(t|d)$ is estimated using the maximum likelihood Eq.12. Similarly to the previous discussion for extending Dirichlet smoothing, we also refine the probability for a term $t \in d_q$, $P_\lambda(t|d_q)$ which is defined as:

$$P_\lambda(t|d_q) = \begin{cases} (1-\lambda)\frac{\#(t;d)}{|d_q|} + \lambda P(t|C) & \text{if } t \in d \\ \\ (1-\lambda)\frac{\#(t^*;d).Sim(t,t^*)}{|d_q|} + \lambda P(t|C) & \text{if } t \notin d \end{cases} \tag{16}$$

3.5 Term Similarity

We only focus, in this work, on the hierarchical relation or specific-generic relations between terms. We make the assumption that a term t is semantically related to a term t', iff t' is a descendant of t in the term hierarchy within an external knowledge K. Assume a query term t, t' refers to a document term, and the vocabulary V. We define the semantic similarity function $Sim(t,t')$ as follows, $Sim : V \times V \rightarrow [0,1]$:

$$\forall t, t' \in V, 0 \leq Sim(t,t') \leq 1 \tag{17}$$

1. $Sim(t,t') = 0$, if t and t' are not semantically related, and $t \neq t'$.
2. $Sim(t,t') < 1$, if t' is a descendant of t in the term hierarchy in K, and $t \neq t'$.
3. $Sim(t,t') = 1$, if $t = t'$.

[3] The knowledge refers to the semantic similarity between terms.

The similarity Sim denotes the strength of the similarity between the two terms (the larger the value, the higher the similarity between these two terms). We propose to use a lightweight way to calculate the semantic similarity between terms. Our semantic similarity relies on a term hierarchy in an external knowledge. The similarity between two terms t and t' is the inverse of their distance, denoted $distance(t, t')$, between these two terms. We use the path length or the number of links in the hierarchy between two terms as distance [19].

The similarity score is inversely proportional to the number of nodes along the shortest path between the two terms. The shortest possible path occurs when the two terms are directly linked. Thus, the maximum similarity value is 1:

$$Sim(t, t') = \frac{1}{distance(t, t')}, distance(t, t') > 0 \tag{18}$$

4 Experimental Setup

4.1 Documents and Queries

Conceptual indexing is the process of mapping text into concepts[4] of an *external resource*. Therefore, it needs a resource out of documents and queries which contains concepts, their relations, and other information about them. In our study, we use concepts as indexing terms i.e. documents and queries are represented by means of concepts rather than words.

Documents and queries are mapped into UMLS[5] concepts using MetaMap [1]. UMLS is a multi-source knowledge base in the medical domain, whereas, MetaMap is a tool for mapping text to UMLS concepts. Using concepts allows us to investigate the semantic relations between concepts, so it allows to build our concepts similarity values. We only consider, the hierarchical relations or specific-generic relations(ISA) between concepts from the different UMLS concept hierarchies. We define general concepts which are internal nodes in a concept hierarchy, or nodes which have at least one child. Returning to the example about term specificity in the introduction, the general concept "*Lymphocyte*" has two children "*B-Cell*" and "*T-Cell*". Then, when a user query contains the term "*Lymphocyte*", then, a document talking about "*B-Cell*" or "*T-Cell*" is retrieved using our approach. Therefore, general concepts have the potential to be linked, in the case of mismatch, to a descendant concept from a document using our extended matching model.

4.2 Corpora

Five corpora from CLEF[6] are used. Table 1 shows some statistics about them.

[4] "Concepts can be defined as human understandable unique abstract notions independent from any direct material support, independent from any language or information representation, and used to organize perception and knowledge [5]".

[5] Unified Medical Language System (http://www.nlm.nih.gov/research/umls/).

[6] www.clef-initiative.eu

– Image2010, Image2011, Image2012: contain short medical documents and queries.
– Case2011, Case2012: contain long medical documents and queries.

Table 1. Corpora statistics. *avdl* and *avql* are average length of documents and queries. Number of general concepts inside the queries.

Corpus	**#d**	**#q**	**avdl** (words)	**avql** (words)	**Number of Concepts in the Queries**	**Number of General Concepts**
Image2010	77495	16	62.12	3.81	186	109
Image2011	230088	30	44.83	4.0	374	198
Image2012	306530	22	47.16	3.55	204	132
Case2011	55634	10	2594.5	19.7	516	219
Case2012	74654	26	2570.72	24.35	1472	519

4.3 Results

All the experiments are conducted using the XIOTA engine [4]. The performance is measured by Mean Average Precision (MAP). The approaches used for experiments are as follows:

– DIR-BL (baseline): language model with Dirichlet smoothing.
– JM-BL (baseline): language model with Jelinek-Mercer smoothing.
– DIR-TM: translation model using Dirichlet smoothing [9].
– JM-TM: translation model using Jelinek-Mercer smoothing [9].
– DIR-CS: our extended Dirichlet smoothing after integrating the concept similarity.
– JM-CS: our extended Jelinek-Mercer smoothing after integrating the concept similarity.

Results of our extended language models are summarized in Table 2. We first observe a consistent performance improvement achieved over ordinary smoothing methods for our five target collections, which confirms our belief that integrating hierarchical relations from an external resource improves relevance model estimation. Second, the improvement occurs in the studied collection is independent to the length of documents and queries in these collections. It seems to be similar for both types of collection: 1) short documents and short queries, 2) long documents and long queries. Finally, the improvement in the two collections: Image2010 and Case2012 is not statistically significant because:

– Image2010: general concepts present in a limited number of queries and not well distributed overall the collection queries.
– Case2012: the rate of general concepts is not high enough comparing with other collections to significantly affect the improvement.

In nutshell, the improvement is related to the rate of general concepts and their distribution inside queries.

We now check how our extended models performs as comparing with the statistical translation models. Table 2 shows the results for (DIR-CS,JM-CS) and (DIR-TM,JM-TM) methods. Comparing the columns (DIR-CS,JM-CS) and (DIR-TM,JM-TM) indicates that our extended models (DIR-CS,JM-CS) are, in most cases, better than statistical translation models (DIR-TM,JM-TM). Significant tests using Fishers Randomization [18] show that our extended models are statistically better than ordinary language models in five cases, whereas statistical translation models are statistically better in only three cases.

Table 2. MAP of Extended Dirichlet smoothing and Extended Jelinek-Mercer smoothing after integrating concept similarity. The gain is the improvement obtained by our approach over ordinary language models. † indicates a statistically significant improvement in over ordinary language models using Fishers Randomization test with $p < 0.05$.

Corpus	**DIR-BL**	**DIR-TM**	**DIR-CS**	**Gain**	**JM-BL**	**JM-TM**	**JM-CS**	**Gain**
Image2010	0.2571	0.2868	0.3049	+19%	0.2494	0.3008	0.3023	+21%
Image2011	0.1439	0.1550†	0.1559†	+7%	0.1641	0.1759	0.1757†	+7%
Image2012	0.1039	0.1188	0.1177	+12%	0.1068	0.1102	0.1186†	+11%
Case2011	0.1103	0.1212†	0.1192†	+8%	0.1480	0.1585†	0.1590†	+7%
Case2012	0.1788	0.1462	0.1861	+4%	0.1871	0.1873	0.1961	+5%

5 Conclusion and Future Work

We propose, in this paper, a model to exploit semantic relations between indexing terms in order to overcome the term mismatch problem. The proposed approach is based on modifying documents according to a given user query and some knowledge about semantic term relations. We extend the document by query terms which are not in the document but they are semantically related to at least one document term. We then integrate the modified document into two smoothing methods from language models: Dirichlet and Jelinek-Mercer. We only consider hierarchical relations between concepts in our similarity measure. Our experimental results indicate that our extended models are statistically better than exact match approaches, and in most cases better than translation models in retrieval performance. This improvement is independent of the length of documents and queries within the tested collections, but it is related to the rate of general terms and their distribution inside queries. We believe that our extension is suitable to integrate any other type of mutual information between indexing terms.

For future work, we plan to validate our extension using other types of relations between terms rather than hierarchical relations. In addition, we plan to apply our extension to other domains rather than the medical domain and other test collections like TREC.

References

1. Alan R Aronson. Metamap: Mapping text to the umls metathesaurus. 2006.
2. Michael Bendersky and W. Bruce Croft. Discovering key concepts in verbose queries. SIGIR '08, pages 491–498, New York, NY, USA, 2008. ACM.
3. Adam Berger and John Lafferty. Information retrieval as statistical translation. SIGIR '99, pages 222–229, New York, NY, USA, 1999. ACM.
4. Jean-Pierre Chevallet. X-iota: An open xml framework for ir experimentation. volume 3411 of *Lecture Notes in Computer Science*. Springer, 2005.
5. Jean-Pierre Chevallet, Joo-Hwee Lim, and Diem Thi Hoang Le. Domain knowledge conceptual inter-media indexing: Application to multilingual multimedia medical reports. CIKM '07, pages 495–504. ACM, 2007.
6. Fabio Crestani. Exploiting the similarity of non-matching terms at retrieval time. *Journal of Information Retrieval*, 2:25–45, 2000.
7. Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6):391–407, 1990.
8. Yufeng Jing and W. Bruce Croft. An association thesaurus for information retrieval. pages 146–160, 1994.
9. Maryam Karimzadehgan and C.X. Zhai. Estimation of statistical translation models based on mutual information for ad hoc information retrieval. ACM, 2010.
10. Robert Krovetz. Viewing morphology as an inference process. pages 191–202. ACM Press, 1993.
11. Victor Lavrenko and W. Bruce Croft. Relevance based language models. SIGIR '01, pages 120–127, New York, NY, USA, 2001. ACM.
12. Jimmy Lin and Dina Demner-Fushman. The role of knowledge in conceptual retrieval: a study in the domain of clinical medicine. SIGIR '06, 2006.
13. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.
14. Fuchun Peng, Nawaaz Ahmed, Xin Li, and Yumao Lu. Context sensitive stemming for web search. SIGIR '07, pages 639–646, New York, NY, USA, 2007. ACM.
15. Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. SIGIR '98, pages 275–281. ACM, 1998.
16. M. F. Porter. Readings in information retrieval. chapter An algorithm for suffix stripping, pages 313–316. Morgan Kaufmann Publishers Inc., 1997.
17. Gerard Salton, editor. *The SMART Retrieval System - Experiments in Automatic Document Processing*. Prentice Hall, Englewood, Cliffs, New Jersey, 1971.
18. Mark D. Smucker, James Allan, and Ben Carterette. A comparison of statistical significance tests for information retrieval evaluation. CIKM '07. ACM, 2007.
19. Dominic Widdows. *Geometry and Meaning*. Center for the Study of Language and Inf, November 2004.
20. ChengXiang Zhai. *Statistical Language Models for Information Retrieval*. Now Publishers Inc., Hanover, MA, USA, 2008.
21. Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to information retrieval. 22(2):179–214, April 2004.