Author-produced version of the paper presented at 6ème atelier Recherche d'Information SEmantique RISE, SDNRI 2014, March 18th, 2014, Nancy, France.
The original publication is available at https://sites.google.com/site/frenchsemanticir/documents

Integrating Terms Hierarchy into Dirichlet Language Model

Mohannad ALMasri*, KianLam Tan*, Jean-Pierre Chevallet**, Philippe Mulhem***, and Catherine Berrut****

* Université de Grenoble, ** UPMF-Grenoble 2, *** Centre National de la Recherche Scientifique, **** UJF-Grenoble 1
LIG laboratory, MRIM group, Grenoble, France {mohannad.almasri, kian-lam.tan,jean-pierre.chevallet,philippe.mulhem,catherine.berrut}@imag.fr

Abstract. Most Information retrieval systems (IRSs) use the intersection between document and query in order to retrieve relevant documents to a given query. Term mismatch problem appears when users use different terms form terms used in the index to express their needs. Indexing term specificity is one face of term mismatch problem where a user query contains more general indexing terms from terms in the index. In this paper, we present an approach to capture the specificity of terms by incorporating the hierarchical information between indexing terms into a Language Model. Experiments on different CLEF corpora from the medical domain show an improvement in retrieval performance. We show that this improvement is independent to the length of documents and queries within the tested collection.

1 Introduction

Specificity is a semantic property that can be applied to index terms: an indexing term[1] is more or less specific as its meaning is more or less detailed and precise. For instance, in the medical domain, the terms "*B-Cell*" and "*T-Cell*" are more specific than "*Lymphocyte*", or in other words, we say that "*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 talks about "*B-Cell*" or "*T-Cell*" is relevant to this query. Another example from the same domain, documents talk about "*Veins*" or "*Arteries*" are relevant to the query "*Blood Vessel*", where "*Veins*" and "*Arteries*" are types of "*Blood Vessel*". A retrieval model that depends on the intersection between document and query cannot capture this kind of relation between indexing terms. In order to take these hierarchical relations between query and document terms, we should incorporate them in the retrieval model. We need an external knowledge or resource [2] further from query and document to identify these hierarchical relations between terms.

[1] Indexing terms differ from system to another, so it can be : word, noun phrase, n-gram, or concept [5].
[2] like thesaurus or ontology

Classical indexing techniques represent documents and queries as a bag of words or phrases without taking into account the semantics, the meaning or the correlation between these words. The main disadvantage of these techniques is that they depend on the text signal, and not on the meaning [5, 12]. For example, in the medical domain, the two phrases "Atrial Fibrillation" and "Auricular Fibrillation" have the same meaning. However, when we use phrases to represent a document and a query, if one phrase appears in a document and a different one appears in a query, that leads to mismatch problem. So over the last 20 years, several approaches attempted to use available knowledge bases and natural language processing techniques in order to overcome this problem and produce more meaningful answers [1]. These approaches represent documents and queries by means of concepts. This representation is obtained using conceptual indexing. Conceptual indexing is the process of mapping text into the concepts of an *external resource*. Therefore, it needs a resource out of documents and queries which contains concepts and the information about them. In our study, we use concepts as indexing terms.

In this paper, we consider the problem of concepts specificity within in the Language Model (LM). Language Model for IR has been proven to be a very effective method for text retrieval [15, 20]. The extension that we propose in this paper is to integrate concept hierarchy into the Dirichlet language model. This extension is easily applied to other smoothing methods. Our proposed method has the following advantages: a) it is easy and simple to generate concept similarity based on the hierarchy of concepts from an external resource b) we propose a light weight integration in the Dirichlet smoothing that improves the retrieval performance. The rest of this paper is organized as follows. Firstly, we present the problem of term mismatch in Section 2. Then, we discuss several approaches to solve the problem of term intersection in Section 3 followed by our approach in Sections 4. Finally, we conclude our results and present the future work in Sections 5 and 6.

2 Term Mismatch Problem

Several techniques [13] have been proposed to tackle term mismatch problem. Among these techniques: relevance feedback [11, 17], dimension reduction [16, 10, 7, 2, 8], and statistical translation model [3, 9].

Relevance feedback involves the user in the IR process in order to improve the final result. 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 made by adding terms extracted from the selected documents. Implicit feedback incorporates user behavior like clicks, in order to predict relevant documents to reformulate the query, while blind feedback provides a method for automatic local analysis. Blind feedback automates the manual part of the Rocchio algorithm without an extended interaction with the user. This method

performs normal retrieval and finds an initial set of relevant documents and makes the assumption that the top k ranked documents are the most relevant. Lavrenko and Croft [11] proposed an approach to estimate a relevance model with no training data. The main problem for implicit and explicit relevance feedback is that they should rely on accurate ways of finding term relation in order to avoid the problem of query drift.

Dimension reduction is the process of reducing the number of data dimensions that represents a query and a document in cases where the query and the document refer to the same concept but using different terms. This can be achieved by using thesaurus [8], concept based approach [2], stemming [16, 10], and latent semantic indexing [7]. All these techniques proposed different strategies to reduce the chances that the query and document refer to the same concept but using different terms. In later development, Peng et al.[14] performed stemming 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[18] and Krovetz[10]. Deerwester et al.[7] proposed to solve the dimension reduction by representing the terms and the documents in a latent semantic space where the terms that are similar in the space tend to be the terms that not only co-occur in the documents, but also appear in similar contexts.

Statistical Translation Model is a model where all of the translations are generated on the basis of a statistical models. This idea is based on information theory where a model estimates the probability of translate a document to a user query according to the probability distribution $P(u|v)$, which gives the probability that a word, v can be semantically translated to a word, u [19, 3]. Unfortunately, Statistical Translation Model requires training data and some relevant query-document pairs where the documents are relevant to the query.

3 Exploiting term similarity

Most of the approaches to solve the problem of term mismatch described in Section 2 faced the same problem: how to select the best term and assign the best weight to the corresponding term?. No single solution has been proved to be the best.

Some approaches have been proposed using LM such as the work of Karimzadehgan and Zhai [9], Berger and Lafferty[3] who use Statistical Translation Model. The main difference between these two works is that Berger and Lafferty try to identify the most important concepts apears in a verbose query [3] while Karimzadehgan and Zhai used mutual information to generate term links[3] [9].

In some ways, we can consider that the proposed approach [9, 3] are related to the proposition from Crestani [6] where the idea is to consider the similarity between each query term and all document terms. The results obtained by Karimzadehgan and Zhai [9] showed that integrating the term similarity and LM is more efficient and more effective than the existing approaches in information retrieval.

[3] Term links refer to the relationship between two terms in a vocabulary

Author-produced version of the paper presented at 6ème atelier Recherche d'Information SEmantique RISE, SDNRI 2014, March 18th, 2014, Nancy, France.
The original publication is available at https://sites.google.com/site/frenchsemanticir/documents

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 term, 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 1) the normalization of the mutual information is rather artificial and requires a parameter to control the effect of the self-translation, and 2) the regularization of the initial transition probabilities may look uncertain.

4 Proposed Approach

Our model uses concepts as indexing terms. In other words, queries and documents are represented by concepts. Then, we use hierarchical relations from an external resource to build the Concept Similarity Matrix. This matrix contains semantic similarities between each two concepts computed from an external resource. Our goal is to integrate the Concept Similarity Matrix into LM in order to overcome the mismatch problem. After the reviews of Crestani [6], Karimzadehgan and Zhai [9] and Zhai [19], we propose the approach as shown below:

- In the case that there is a query concept does not appear in the document(Mismatch): we consider the most similar document concept to this query concept in the matching process. We use concept similarity matrix to find the most similar concept.
- We propose to exploit hierarchical relations between concepts which is defined in the external knowledge to define the semantic link between concepts rather than probability approaches in order to avoid the problem of self-translation Karimzadehgan and Zhai [9].

In order to build the Concept Similarity Matrix, we find the links between all the vocabulary V which is the set of all concepts. We make the assumption that the two concepts are considered to be linked to each other if both concepts belong to the same hierarchy in the external resource. Assume a query concept c, and c' refers to a document concept:

$$c, c' \in V, 0 \leq Sim(c, c') \leq 1 \tag{1}$$

1. $Sim(c, c') = 0$, there is no link between the concept c and c'
2. $Sim(c, c') < 1$, there is a link between the concept c and c'
3. $Sim(c, c') = 1$, there is an exact match between the concept c and c'

4.1 Extended Dirichlet Smoothing

The LM approach in IR is proposed by Ponte and Croft [15]. The basic idea of LM is to assume that a query q, which is generated by a probabilistic model based on a document d, as shown below:

Author-produced version of the paper presented at 6ème atelier Recherche d'Information SEmantique RISE, SDNRI 2014, March 18th, 2014, Nancy, France.
The original publication is available at https://sites.google.com/site/frenchsemanticir/documents

$$P(d|q) \propto P(q|d).P(d) \tag{2}$$

$\propto$ means that the two sides give the same ranking. $P(q|d)$ the query likelihood for the given document d matches with the query q. If we consider that every document is equally relevant to any other query, then we can discard $P(d)$ and we can rewrite the formula after adding the log function as:

$$logP(d|q) = \sum_{c \in V} \#(c;q).logP(c|d) \tag{3}$$

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

$$P_{ml}(c|d) = \frac{\#(c;d)}{|d|} \tag{4}$$

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

Basically, Dirichlet [21] is one of the smoothing technique based on the principle of adding an extra pseudo concept frequency: $\mu P(c|C)$. The Dirichlet smoothing is obtained by taking into account the extra pseudo concept frequency distribution:

$$P_{\mu}(c|d) = \frac{\#(c;d) + \mu P(c|C)}{\sum_c \#(c;d) + \mu} \tag{5}$$

where C is the whole collection. The main idea of this proposal is to integrate links between concepts which are represented by Concept Similarity Matrix into the current Dirichlet formula. First, we assume that for a query concept $c \in q, c \notin d$, there is a document concept $c' \in d$ can play its role during the matching process. More specifically, we consider that if c does not occur in the initial document d but occurs in the *document* d_{ext}, which is the result of extending d according to the query and some knowledge [4], the probability of the concept c' is defined according to the extended document d_{ext}.

The knowledge provides a similarity function $Sim : V \times V \rightarrow [0, 1]$, that denotes the strength of the similarity between the two concepts (the larger the value, the higher the similarity between these two concepts). We propose that: $\forall c, c' \in V, Sim(c, c') = 1$ if exact matching between c with c', and $\forall c, c' \in V, Sim(c, c') = 0$ if c is not at all semantically related to c'.

In order to avoid any complex extension, we assume that a query concept c, must only impact occurrences of one document concept, so:

1. If a query concept c occurs in a document d, then the concept will not change the length of the document.

[4] The knowledge refers to the Concept Similarity Matrix

2. If a query concept c does not occur in a document d but the concept c contains a link with c' (concept from document), then we define: :

$$c^* = argmax_{c' \in d} Sim(c, c')$$

as the concept from the document will serve as the basic count of the pseudo occurrences of c in d:

$$\#(c^*; d).Sim(c, c^*)$$

this pseudo occurrences of the concept c are then included into the size of the extended document.

3. If a query concept c does not occur in the document and does not show any link to the document concepts , then this concept will not change the length of the document as the first case.

According to the previous three cases, the expression of $|d_{ext}|$ can be done:

$$|d_{ext}| = |d| + \sum_{c \in q} \#(c^*; d).Sim(c, c^*) \tag{6}$$

Note that we propose to extend the document according to the query. We extend the document by query concepts which are not in the document but they are linked to at least on document concept. Now, the extended Dirichlet Smoothing leads to the following probability for the concept c of the vocabulary V in the extended document d_{ext} according to a query q, and note that $p_\mu(c|d_{ext})$ is defined as:

1. if $c \in d \cap q$:

$$P_\mu(c|d_{ext}) = \frac{\#(c; d) + \mu P(c|C)}{|d_{ext}| + \mu} \tag{7}$$

2. $c \notin d \cap q$ and if $\exists c^* \in d \setminus q; Sim(c, c^*) \neq 0$:

$$P_\mu(c|d_{ext}) = \frac{\#(c^*; d).Sim(c, c^*) + \mu P(c^*|C)}{|d_{ext}| + \mu} \tag{8}$$

with $c^* = argmax_{c' \in d} Sim(c, c')$.

3. $c \notin d \cap q$ and if $\nexists c^* \in d \setminus q; Sim(c, c^*) \neq 0$

$$P_\mu(c|d_{ext}) = \frac{\#(c; d) + \mu P(c|C)}{|d_{ext}| + \mu} \tag{9}$$

with $c^* = argmax_{c' \in d} Sim(c, c')$.

In the specific case where all the query concepts occur in the document, we have $|d_{ext}| = |d|$, and that leads to $p_\mu(c|d) = p_\mu(c|d_{ext})$.

4.2 Concept Similarity Matrix

We propose to use a lightweight way to build the Concept Similarity Matrix using the concept hierarchy from an external resource. The similarity between two concepts is the inverse of a distance between these two concepts in the concept hierarchy [18]. We use the path length or the number of links in the hierarchy between two concepts as distance.

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

$$Sim(c, c') = \frac{1}{distance(c, c')}, distance(c, c') > 0 \quad (10)$$

We also tried other similarity metrics like Leacock and Resnik but we obtained best performance improvement using this path metric.

5 Experimental Setup

5.1 Indexing Terms

Documents and queries is mapped to UMLS concepts using MetaMap. 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 Matrix. To build this matrix, we only consider, the ISA relations between concepts from the different UMLS concept hierarchies. If we have two concepts in multiple concept hierarchies we consider the shortest path.

5.2 Corpora

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

- Image10, Image11, Image12: contain short medical documents and queries.
- Case2011, Case2012: contain long medical documents and queries.

5.3 Results

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

- DIR-BL (baseline): language model with Dirichlet smoothing.
- DIR-CSM: Extended Dirichlet smoothing after integrating the Concepts Similarity Matrix(CSM).

Table 1. Corpora statistics. *avdl* and *avql* are average length of documents and queries. Number of general concepts inside the queries, or in other words the number of concepts which has the potential to be subsumed at matching time.

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
Case2011	55634	10	2594.5	19.7	516	219
Image2012	306530	22	47.16	3.55	204	132
Case2012	74654	26	2570.72	24.35	1472	519

Results of our Dirichlet smoothing extension are summarized in Table 2. We first observe the consistent performance improvement achieved 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 the length of these collection. It seems to be similar for both types of collection: 1) short documents and short queries, 2) long documents and long queries.

Table 2. MAP of Extended Dirichlet smoothing after integrating Concept Similarity Matrix: our approach outperforms the baseline result for all studied collections. The gain obtained could be related to the number of general concepts inside the queries, or in other words the number of concepts which has the potential to be subsumed at matching time.

Corpus	General Concepts Rate	**DIR-BL**	**DIR-CSM**	**Gain**
Image2010	59%	0.2571	0.3049	+19%
Image2011	53%	0.1439	0.1540	+7%
Case2011	42%	0.1493	0.1597	+7%
Image2012	65%	0.1039	0.1131	+9%
Case2012	35%	0.1788	0.1861	+4%

Table 3 show some statistics about our three cases in the extended model during the matching between documents and queries. We see in this table: 1) the number of shared between documents and queries. 2) the number document concepts linked to unmatched query concept.3) the number of document concepts which they can not be linked to a query concepts. These number are over all queries and documents in the studied collections.

6 Conclusions

We present a model to exploit the indexing term hierarchy in order to capture the specificity of indexing terms during the retrieval time. Our experimental re-

Author-produced version of the paper presented at 6ème atelier Recherche d'Information SEmantique RISE, SDNRI 2014, March 18th, 2014, Nancy, France.
The original publication is available at https://sites.google.com/site/frenchsemanticir/documents

Table 3. Statistics during the matching between documents and queries: 1) number of shared between documents and queries. 2) number document concepts linked to unmatched query concept.3) number of document concepts which they can not be linked to a query concepts.

Corpus	#Shared	#Linked	#Not linked
Image2010	2,138,561	668,148	11,607,361
Image2011	2,492,692	1,275,058	82,285,162
Case2011	6,725,714	932,321	21,049,109
Image2012	2,874,031	1,556,122	91,169,348
Case2012	27,940,254	3,681,351	78,255,835

sults indicate that the proposed approach to extend Dirichlet smoothing using Concept Similarity Matrix based on hierarchical information from an external resource in the medical domain is more effective than the term intersection approach. This extension is suitable for any situation where such a kind of this mutual information between indexing terms is available. For future work, we would like to validate our extension using mutual information between indexing terms extracted from other external resource and maybe in different ways rather than hierarchical relations. In addition, we think that with more mutual information we will have a higher degree of knowledge to build the link between two indexing terms.

References

1. Mustapha Baziz, Mohand Boughanem, and Nathalie Aussenac-Gilles. Conceptual indexing based on document content representation. CoLIS'05, 2005.
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*, pages 263–280. Springer Berlin Heidelberg, 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. 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 ChengXiang Zhai. Estimation of statistical translation models based on mutual information for ad hoc information retrieval. pages 323–330. 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, pages 99–106, 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. Dominic Widdows. *Geometry and Meaning*. Center for the Study of Language and Inf, November 2004.
19. ChengXiang Zhai. *Statistical Language Models for Information Retrieval*. Now Publishers Inc., Hanover, MA, USA, 2008.
20. Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to information retrieval. 22(2):179–214, April 2004.
21. Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to information retrieval. 22(2):179–214, April 2004.