Concept Tree Based Information Retrieval Model

Yuan Chunyan
Institute of Civil Engineering, Chang'an University, Chang'an University, Shanxi, Xi'an, China 710061
Email: jancyer@163.com

***Abstract*—This paper proposes a novel concept-based query expansion technique named Markov concept tree model (MCTM), discovering term relationship through the concept tree deduced by term markov network. We address two important issues for query expansion: the selection and the weighting of expansion search terms. In contrast to earlier methods, queries are expanded by adding those terms that are most similar to the concept of the query, rather than selecting terms that are similar to a signal query terms. Utilizing Markov network which is constructed according to the co-occurrence information of the terms in collection, it generate concept tree for each original query term, remove the redundant and irrelevant nodes in concept tree, then adjust the weight of original query and the weight of expansion term based on a pruning algorithm. We use this model for query expansion and evaluate the effectiveness of the model by examining the accuracy and robustness of the expansion methods, Compared with the baseline model, the experiments on standard dataset reveal that this method can achieve a better query quality.**

***Index Terms*—Concept Tree; Markov Networks; Query Expansion; Information Retrieval**

I. INTRODUCTION

Since the earliest days of IR, researchers noted the potential defects of keyword retrieval, such as synonymy, polysemy, hyponymy or anaphora. Although in principle these linguistic phenomena should be taken into account in order to obtain high retrieval relevance, the lack of algorithmic models prohibited any systematic study of the effect of this phenomena in retrieval. Instead, researchers resorted to distributional semantic models to try to improve retrieval relevance, and overcome the brittleness of keyword matches.

Most research concentrated on Query Expansion (QE) methods, which typically analyze term co-occurrence statistics in the corpus and in the highest scored documents for the original query in order to select terms for expanding the query terms [1]. Such techniques are used to augment the original query to produce a representation that better reflects the underlying information need. Current, the most commonly methods used to treatment the query are relevance feedback [2], query reconstruction and automatic query expansion [3]. According to document sets, automatic query expansion technique divided into local analysis method and global analysis method. As an early appeared query expansion method, global analysis method has been widely used.

The traditional global analysis method didn't take the indirect relationship between terms into account. And in traditional approaches of query expansion, a term will be chosen if it associates strongly with a certain query term in a given query. However, the entire query concept is rarely taken into account, in reality, terms which are strongly similar to the query topic are more important and benefit to improve efficiency. Therefore, such kind of terms should be added to expand original query.

Recently, a Markov network model (MNM) for information retrieval was proposed that goes beyond the simplistic bag of words assumption that underlies BM25 [4]. The Markov network retrieval algorithm, in particular, has been shown to be ideal for concept-based information retrieval. The work in Zuo [5] first proposes and implements an information retrieval model based on Markov network. Markov Network is an undirected graphical network that is capable of efficiently representing relevance in knowledge and can be easily gotten from training data. They construct the Markov network to represent the relationships between terms and the relationships between documents learned from training corpus, then do query expansion and re-compute the similarity between documents and queries. Experiments show the result is promising. In Cao [6] they consider the retrieval model as an inference process. Its isotropy take the query as the evidence source, the relevance document is looked upon as active document, improving the performance of information retrieval by adding useful information into retrieval process. Making use of edges without the direction in the network, activating the words closely related with the query as the additional evidential sources into the model. All the models ignore the dependence between query terms, but it is unrealistic.

Associative models consider relationships between terms in addition to the terms themselves. They have been extensively considered and studied for information retrieval, e.g. by Stiles [7], van Rijsbergen [8] and Salton & Buckley [9] among many others. There are many lexical and semantic relations that may be considered for associating a pair of terms. For example: stemming, based on common morphology; synonymy, where aspects of meaning are shared; co occurrence, in which both words tend to appear together; and general association, where a person is likely to give one word as a free-association response to the other.

Each relation may be thought of as an inference step, in which a source word v has some property R(v), and a new word w can be inferred to have the property value R(w) with probability PR(w|v), based on their shared

© 2014 ACADEMY PUBLISHER
doi:10.4304/jmm.9.5.652-659

relation. For example, if vis the word ‘matrix’ and the property R(.) is ‘relevancy to a query’, then one possible way to calculate PR (w|v) is based on co-occurrence, so that, for example, the term ‘row’ also has some measure of relevance. Note that this is not symmetric: ‘row’ having more senses and being more common, it is less likely to imply relevance of ‘matrix’, unless another term is also present for context, such as ‘column’.

While lexical and semantic relations may be useful individually, it is important to consider how they may be used in combination. One reason for this is a common problem in language processing called sparsity: for co occurrence relations for example, even with a huge corpus, we only have reliable co-occurrence data for a fraction of all potential term pairs. External semantic resources such as Word Net or stemming dictionaries supply a broad set of terms but are limited in the depth and currency of their vocabulary. By combining multiple relations into chains of inference, we can help bridge the gaps that exist in the data.

A second reason is that the various relations between words represent potentially complimentary sources of evidence that may help to distinguish and disambiguate terms. For example, if ‘bank’ and ‘merger’ are known to be relevant to a query, then the following inference chains would provide evidence that ‘negotiations’ may also be relevant:

1. bank →agreement (C) →negotiate (C) →negotiations (M)
2. merger →talks (C) →negotiations (S)

where C, S, and M represent co-occurrence, synonymy, and morphology relations respectively. Note that chains can emphasize different types of evidence at different walk stages. In the above example, co-occurring terms are found first, followed by their synonyms or stems.

This paper proposes an concept tree information retrieval model based on Markov network, our work differs from the related work in one or more of the following aspects: first, we used the Markov network model to express the indirect relationship between terms. Moreover, we adjusted the contribution of the initial query and select the most appropriate expansion terms association with the entire query concept according to a pruning algorithm. This is done using query concept tree extracted from the text collection, selecting expanded term relies on the overall similarity with the query concepts rather than the similarity with a signal query term. We focus our attention on the problem of generating concept tree and pruning algorithm. Our approach is to provide high level suggestions for expanding the original user query with additional context. Especially in the Multi-keyword query, numbers of query terms are semantically related; we can reduce the possibility of ambiguity, and improve the precision.

In this paper, we first give a brief introduction about previous work. Then present a construction method that allows getting a concept tree from Markov network. Section 3 is a core part; we introduce the detailed steps of the pruning algorithm, and describe a probabilistic query expansion and weighting model through the Markov concept tree. After describing our test setting, some results of experiments carried out with five standard test collections are presented in section IV. Finally, we conclude with the main findings and point out further research and possible applications of the methods presented.

II. PRELIMINAIES

Markov Network is an undirected graphical network [10] that is capable of efficiently representing relevance in knowledge and can be easily gotten from training data with strong learning and inferring capability. Since non-direction of the edge in Markov network, construct the Markov network is much easier than the discovery of Bayesian networks. Using the non-direction edge to explain the semantic relationship between items in information retrieval is more intuitive and appropriate. In addition, Markov network have powerful learning ability to analyze the instance dataset.

Markov network can be represented by the graph G = (V, E), V is the set of vertices is the set of edges, E = {(xi, xj) | xi≠xj, xi, xj∈V}, The nodes in the graph represent the random variables and the edges define the independence semantics of the distribution. The graph satisfies the Markov property, which states that a node is independent of all of its non-neighboring nodes given observed values for its neighbors. That is p(xi | xj)=p(xi | xj,(xi, xj)∈E).

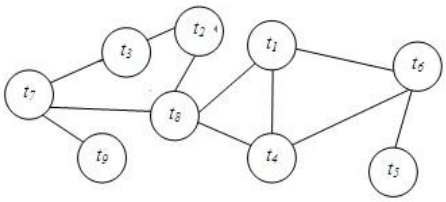


Figure 1. Markov network structure

In the case of query expansion, the target terms are potential expansion terms, the source terms are initial query, and the labels are probabilities of relevance. We then define a random process to propagate the label information through the graph. The stationary distribution of this process gives us a probability distribution over expansion terms. Figure 1 shows a portion of a term network for the query ‘Parkinson’s disease’. Solid connections denote explicit term associations, while the dashed line shows an implied connection inferred between ‘brain’ and ‘dopamine’ based on a short chain through the shared node ‘Parkinson’s disease’. We extend earlier work by Lafferty and Zhai [11] on using Markov chains for query expansion, by using a more flexible family of random walks similar to that described in Toutanova [12].

The Markov chain approach for modeling term associations is related to previous models based on term clustering and spreading activation networks, both of which have a long history that will only be briefly summarized here.

© 2014 ACADEMY PUBLISHER

Stiles [13] described heuristics for using sets of term associations in improved indexing, and later Quillian [14] proposed a semantic network of concepts for binary relations between words. Gotlieb and Kumar [15] devised a semantic clustering of index terms using maximal complete subgraphs in a term network, although their method's effectiveness was never evaluated for retrieval. Early work in term clustering for query expansion by Sparck Jones [16] focused on constructing a similarity matrix of single index terms before any user queries were submitted. Wong and Raghavan [17] proposed the use of a matrix of term-term associations in ranking document vectors against query vectors, focusing on the special case of term correlation based on co-occurrence. Salton and Buckley [9], van Rijsbergen [8], and many others explored organizing of associations into networks for expanding the search vocabulary. These networks used various node activation heuristics and decay rules that were intuitively plausible but had limited retrieval success. Crestani [18] gives a summary of earlier work on spreading activation networks.

Stationary distributions have been used previously in information retrieval for 'influence weighting' schemes such as PageRank [19] and hub-authority [20] [21], and also for query expansion [11]. Lafferty and Zhai considered a bipartite graph on query terms and documents and calculated an approximate stationary distribution using a random walk. In their scheme, the random walk was defined in terms of words and documents, not words only. Our local co occurrence link is calculated in a similar way, but our random walk framework is more general in that we can use multiple sources of lexical and semantic evidence, not just co-occurrence, with the potential to weight these sources differently at different stages of the walk.

With regard to other query expansion approaches, the idea that we present below of rewarding expansion terms reflecting multiple aspects of the original query was previously noted by Xu and Croft [22] for Local Context Analysis (LCA), which has shown good empirical performance. Their method uses an empirically derived formula to score potential expansion terms that is similar in effect to our probabilistic term scoring. A number of studies have used external resources for query expansion. For example, Voorhees used Wordnet with limited success [23]. Shah and Croft [24] used Wordnet synonyms to perform query expansion for high-precision retrieval, selecting terms with high clarity scores.

In this paper, we pay attention to construct the Markov network of the items; therefore, we will select the appropriate method to measure the relationship between words, so we choose the mutual information after normalized for determining the similarities of all the term pairs (ti, tj). Fig. 1 illustrates the structure, and the value on the edge stands for the similarity of the two connected nodes. In order to reduce the complexity of the Markov network, we assume that only the similarity value is larger than the threshold β (set according to from 0.04 to 0.32), adding an edge between the two items. The general motivation for using a Markov chain on a network of terms is that we want to infer a particular property (the label) of a target word given a set (usually small) of labeled source words.

In contrast to many early spreading activation systems, the Markov chain approach to query expansion is relatively simple and offers a well-motivated probabilistic framework that fits well within the language modeling approach to information retrieval. Moreover, its close relationship with semi-supervised learning [25] means that we may make use of insights from that area to help illuminate the nature of query expansion and learn more robust expansion algorithms.

III. Concept Tree Based Markov Information Retrieval Model

The proposed algorithm is composed of four basic steps. First, build the term Markov network by calculating the similarity between terms; Second, construct concept tree for each original query, each tree is viewed as a concept, each child node is related to their root node with various relationships(such as synonyms, describing the same thing, representing the same concept…), the association is computed through Markov random walk matrix; Third, calculate the similarity between query items according to the association of their concept tree, in order to determine the core query, then remove redundant information node, and adjust the weight of query and the child node of the tree; Fourth, expand the original query with all remaining nodes in their concept trees, compute the similarity between documents and queries.

A. Building Concept Tree

Query expansion technique is a combination of computational linguistics, information science and other technique, expanding the initial user query with additional terms related with initial query to avoid synonyms and missing words problem, cutting off the irrelevant expansion terms to address the words mismatch problem in information retrieval, thus improving retrieval recall and precision. The core issue of query expansion technology is the expansion of word selection and weight [26, 27] distribution.

This article make semantic extension for each query term through the Markov concept tree, all the child node in the concept tree added to the original query as the expansion words, to compensate for the defects of the poor initial query information.

Markov network is a network of interconnected index which closely related between terms, the directly connected index may be synonyms, may be the words to express the same concept or describe the same thing. Markov network would activate the neighbor nodes of the index automatically, even if not directly connected to each other, there may be exist a strong semantic relationship. When retrieval the document, we find the original query item in the Markov cyberspace, then select its closely related (activated) index together as evidence source to calculate the document probability. So in this article we created a concept tree for each query item, each node in the tree has a certain degree of semantic

© 2014 ACADEMY PUBLISHER

association with the root node (the initial query term), we designed the algorithms to create concept tree as follow:

(1) First let p = 1, search the item tj directly connected to the initial query item qm in Markov network space by breadth-first algorithm, then create the concept tree with the root node of qm, let tj be the child of the node q_m, and add the child node to the list Lp and list Lp.

(2) If the list Lp is not null, let pointer q points to the first child node tj;

(3) Remove the node tj from the Lp, and find the item tk connected with tj in Markov network by the breadth-first algorithm; if sim(ti, tj)>σ and $t_k \notin L, t_k \notin L_P$, let tk be the child node of tj, and join in list Lp+1 and Lp list;

(4) the pointer q points to the next node of Lp, if it does not point to end of the list Lp, go to Step 3; otherwise p = p + 1, go to Step 2.

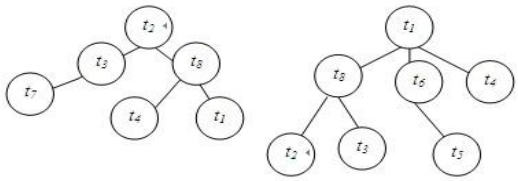


Figure 2. Concept tree

Fig. 2 explain a specific process to create a query concept tree based on the Markov network shown in Fig. 1 Suppose the index item t1 and t2 form a query, and then create two concept trees in Fig. 2. In general, the deeper level the child node in, the weaker correlation with the root node, using Shi [28] the transition probability matrix of the Markov network to update the correlation between each child node and the root of tree, sim (qm, tj) represent the association between the query qm with the index term tj. The parameters of the transition probability matrix of weight coefficients λ set to 0.5, and optimal random walk steps k set to 2.

$$sim(t_i, t_j) = \sum_{t=1}^{k} \lambda^t P_k(t_i, t_j) \quad (1)$$

$$P_1(t_i, t_j) = \sum_{t \in T} P_0(t_i, d_k) P_0(t_k, t_j) \quad (2)$$

$$P_K = P_{K-1} \times P_0 \quad (3)$$

B. Pruning Algorithm for Concept Tree

In the multi-keyword query, each query term plays different role in search task, the weight should also be different. Determining the core query terms by the association degree of the query concept trees, pruning the Redundant node and adjusting the semantic weight of each node in concept tree, making query expansion terms closer to the user's query intent, in order to improve the precision. Our method for pruning is composed of three steps: (1) increasing the weight of core query, (2) pruning the redundant node and (3) decreasing the weight of non-core node. In the following, we discuss each of these 3 steps..

1) Increasing the Weight of Core Query

In the query, if there is some query words semantically related, indicating that the entire query theme will be more inclined to these related query words, in the search task should enhance these words weight. In our approach, using the semantic relevance of the concept tree to calculate the association between the query terms, sim(qm,qn) represent the relationship between the initial query term qm and qn.

$$sim(q_m, q_n) = tree_sim(q_m, q_n) \quad (4)$$

tree_sim(qm,qn) denotes the association between the concept tree of query qm with the concept tree of query qn, values are in the following range, 0≤tree_sim(qm,qn)≤1,the greater value tree_sim(qm,qn) is,the more similar between the two concept tree.

$$tree_sim(q_m, q_n) = sum_co(q_m, q_n) \times sum_co(q_n, q_m) \quad (5)$$

sum_co(qm, qn) represent the degree of overlap information in qm concept tree, between two query concept tree of qm and qn, where sum_co(qm, qn) represent the similarity of qm and tj which can be performed offline according to the formula 1 in precious section.

$$sum_co(q_m, q_n) = \frac{\sum_{t_j \in tree(q_m) \wedge tree(q_n)} sim(q_m, t_j)}{\sum_{t_i \in tree(q_m)} sim(q_m, t_i)} \quad (6)$$

If the correlation between the query terms is great enough, we believe that these query terms are more consistent with the entire query topic, which would play a more important role in search task, it will be defined as the core query, and their weights will be increased. This article will calculate the correlation of each two query pairs, setting the threshold ε, if sim(qm,qn)>ε, set the initial query qm and qn as core query, then update the query weight as follows:

$$weight_adjust(q_m) = (1 + sim(q_m, q_n)) \times weight(q_m) \quad (7)$$

where weight_adjust(qm) represent the weight of query qm after adjustment, and weight(qm) represent the initial weight of query qm.

2) Pruning the Redundant Node

If the value of sim(qm, qn) between two original queries is 0, it indicated that they did not have any semantic relationship. The larger value it is, the more the semantic overlap there will be. We use the whole concept tree instead of single query term as the evidence source to search in document retrieval. So we will adjust the tree to reduce the complexity of the tree, in order to minimize the of noise take into query process, we remove the redundant node between the concept trees. The specific steps are as follow, compare semantic overlap between the query terms. For example, if sum_co(qm, qn)> sum_co(qn, qm), we believe that the semantics of qm is more simple, the more able to express the concept of the

© 2014 ACADEMY PUBLISHER

entire query, so keep the overlapping node in concept tree of qm, but pruning the concept tree of qn,that is to remove the node which appear in two concept tree at the same time, to update the weight of nodes remain in the the qm concept tree moreover.

The weight of the child node should reflect the characteristic of root node, so we take the semantic similarity factor sim(qm,qn) into account.

$$weight_adjust(t_j) = (1 + sim(q_m, q_n)) \times weight(t_j) \times sim(q_m, t_j)$$
$$t_j \in \{\ tree(q_m) \wedge tree(q_n)\ \} \wedge sim(q_m, q_n) > \varepsilon \quad (8)$$

3) Weaken the Non-Core Terms

If the initial query term is polysemous, its expansion terms added into the concept tree are strongly associated with the initial query, but may be not relevant to the entire query topic, that would bring into lots of noise, and would not make precision improved, even may cause the topic drift; the non-overlapping information nodes in the concept tree of query qm and qn produced in the previous step, which is possible be the non-related concepts, we reduce their weights to avoid semantic drift so as to enhance the precision as follows formula:

$$weight_adjust(t_i) = (1 - sim(q_m, q_n)) \times weight(t_i) \times sim(q_m, t_i)$$
$$t_i \notin \{\ tree(q_m) \wedge tree(q_n)\ \} \wedge sim(q_m, q_n) > \varepsilon \quad (9)$$

After above three steps, if sim(qm,qn) value is 0, we believe that there is no semantic relationship between query qm and query qn, treating the node in the query concept tree without any adjustment. Fig. 3 is an example of pruning based on the concept tree shown in Fig. 2.

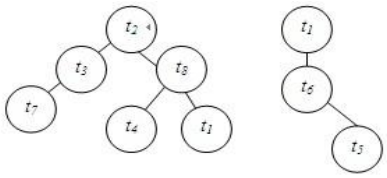


Figure 3. Concept tree after pruning

C. Probabilistic Query Expansion Model

Giving query q, and document set D, we will calculate the probability of p(dj|q) according to the formula 9,where dj is contained in document set D, the range of j is, 1<j<n,and the MT represent the item space of Markov network:

$$p(t_i \mid d_j) \propto \sum p(t_i \mid d_j) p(t_i \mid q) \quad (10)$$

If both the item network and the document network are fixed, p(T|MT) and p(MT) is equal to any dj,

$$p(t_i \mid d_j) \propto \sum p(t_i \mid d_j) p(t_i \mid q) \quad (11)$$

The formula 10 is the general form of the search model algorithm, we can derive formula 11:

$$p(t_i \mid d_j) \propto \sum p(t_i \mid d_j) p(t_i \mid q) \quad (12)$$

We compute the weight [29] as follows:

$$p(t_i \mid d_j) = \frac{idf_i \times dl_j^{\ 2}}{\alpha \sum_{t_i \in d_j} idf_i \times dl_j^{\ 2}} \quad (13)$$

$$p(t_i \mid q) = \frac{\log(tf_i)^2}{\beta \sum_{t_i \in q} \log(tf_i)^2} \quad (14)$$

IV. EXPERIMENTS RESULTS

In order to better understand the strengths and weaknesses of our technique, we evaluate it on a wide range of datasets, five standard test collections shown in Table 1 were used for our experiments. After extracting all the words from the collections and removing stop words, we used Porter Stemmer algorithm to index both queries and documents included by title and body. Table 1 indicates the number of documents, the number of queries, and the number of terms. It can be seen that the Adi collection is rather small and the Cacm collection is sizable as a test collection. The Med, Cran, Cisi collection is of medium size.

TABLE I. EXPERIMENTAL DATASETS

collection	description	Num of doc	Num of query	Num of terms
Adi	Information science	82	35	893
Med	Medical science	1033	30	8702
Cran	Aviation science	1400	225	4110
Cisi	Library science	1460	76	5494
Cacm	Computer science	3024	64	5041

Term weights in both documents and queries are determined according to the log(tf).idf.dl weighting scheme, see also formula (13). The mutual information was used for determine the similarities between all the term pairs in the collections, i.e. build up the Markov network. In addition, the construction method described in section 3 was used to generate the concept tree, Note that this can be achieved very efficiently as the pre-computed the Markov random walk matrix, The entire concept tree are chosen to expand or modify the query according to formulae (3), (4), and (5), Then, for each query, we rank the terms of the collection in decreasing order according to formula (14). The weight of the child node in concept tree cannot be equally treated as root node (initial query term), so introduce the weight adjustment factor η to dynamically adjust the contribution of them:

$$p(d_j \mid q) \propto \sum_{t_i \in q} (\eta\, p(t_i \mid d)\, p(t_i \mid q) + (1-\eta) \sum_{t_m \in tree(t_i)} p(t_m \mid d_j) p(t_m \mid q))$$

The results were evaluated by applying the average precision of a set of queries at three representative recall points 3-avg (0.2, 0.5, 0.8), and eleven representative recall points 11-avg (0,0.1,0.2,…,0.9,1.0). In addition, we now investigate how well our model performs in practice. We provide results for three distinct ranking algorithms: the baseline BM25 [30] model, the basic Markov network model(BMNM) [6], and the Markov concept tree model

© 2014 ACADEMY PUBLISHER

(MCTM),to better understand how each model performs across the various data sets.

Table 2 shows the 11-point average precision for the BM25, BMNM, MCTM algorithms, and table3 shows the 3-point average precision for three algorithms, the improvement also provided as the BM25 for baseline. We observe that BMNM, MCTM algorithms yield better precision than the baseline, MCTM ranking yields the largest gains in average precision in all data sets (Especially increases by around 39% and 27% for the Cisi dataset and the Cacm dataset respectively); and we compare the results between BMNM and MCTM, there is a stable improvement as the BNMN for baseline, which is 5.4%, 4.17%, 5.9% and 6.4% for Adi, Med, Cisi, and Cacm respectively. The figures indicate that our concept tree query expansion method makes a considerable improvement in all collections. That is because its disambiguation scheme can be used to successfully describe user information needs, capturing the concepts related to the user search experience. We also found that the number of expansion terms is really much smaller than the BMNM baseline model.

TABLE II. 3-AVG VALUE OF RESULTS ON DIFFERENT DATA SETS

MODEL	BM25	BMNM	MCTM
Adi	43.09	44.81(3.9%)	49.25(14.3%)
Med	54.32	58.67(11.9%)	61.12(12.5%)
Cran	42.15	42.21(0.14%)	42.32(0.41%)
Cisi	18.17	21.04(15.8%)	22.31 (22.8%)
Cacm	22.94	28.31(23.4%)	30.12(31.2%)

It seems that the improvement increases with the size of the collection. In addition, the improvement increases with the number of additional search terms that expand the original query as long as the collection is large enough. Obviously, the large collection contains more domain knowledge than the small ones. As a consequence, the quality of the concept tree created from the large collection is better than the quality of the concept tree belonging to the small collections. Compared to Cisi there is smaller improvements in Med, it's possible that the feature of dataset Med itself caused, there may be less polysemous, synonyms in the medical science, these words express the semantics are clear. The dataset Cran perform special in the experiments that may be caused by its document specialty. This was true across all collections. The reasons for this require further study, but it suggests that how the evidence is weighted by time-step does indeed matter. In this case, terms that co-occur with terms semantically close to the query appear to be more valuable than terms semantically close to many potential co-occurrence terms.

TABLE III. 11-AVG VALUE OF RESULTS ON DIFFERENT DATA SETS

MODEL	BM25	BMNM	MCTM
Adi	42.09	44.02(7.7%)	48.92(16.2%)
Med	53.08	57.02(7.4%)	58.60(10.4%)
Cran	42.88	42.54(-0.7%)	43.26(0.89%)
Cisi	17.96	23.9(33.1%)	24.97(39%)
Cacm	25.29	29.5(23.6%)	32.12(27%)

In our approach, the parameter η reflect the importance of the extended terms. The higher the value of η is, the greater importance the extended term is, and vice versa, the smaller the value of η, the greater importance of the original query terms. In this experiment, the η value under the optimal results is almost small, η values are less than 0.1 in all the document set. η values in this article seem to follow certain rules, changing with the size of the set of documents. In Fig. 4, we study how the η value affects the usefulness of the concept tree expansion method.

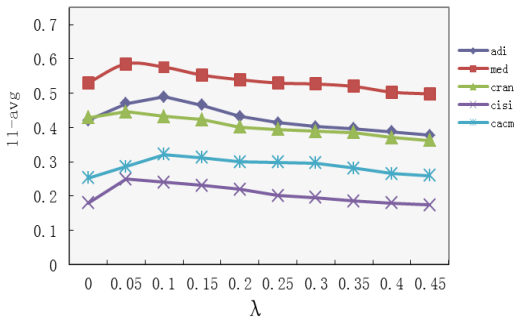


Figure 4. Comparison of different η on each dataset

The results in the previous section indicate that optimization is very important, but unfortunately real applications usually lack training data. In this Section we wanted to study whether the parameters can be carried over from one dataset to the other, and if not, whether the extra terms found by QE would make the system more robust to those sub-optimal parameters.

Fig. 4 includes a range of parameter settings, including defaults, and optimized parameters coming from the same and different datasets. The results show that when the parameters are optimized in other datasets, QE provides improvement with statistical significance in all cases.

Overall, the figure shows that our QE method either improves the results significantly or does not affect performance, and that it provides robustness across different parameter settings, even with suboptimal values.

The depth of concept tree d determines the indirectly relationship between terms, in order to find the most descriptive words to word relevance of different depth, we compared 11-avg value for depth change from 1 to 5 as shown in Fig. 5 To evaluate the efficiencies of different depth of concept tree.

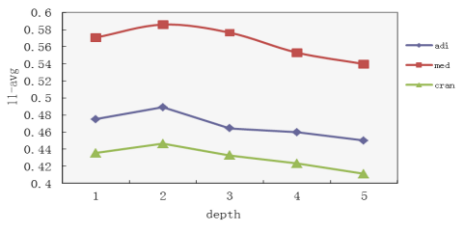


Figure 5. Comparison of different depth of concept tree on Adi, Med, Cran

We can conclude that when depth is 2, all the results reach best. This seems to be the reason information after

© 2014 ACADEMY PUBLISHER

indirect related three time will be deviate from the original intent, only the value of depth in Adi make slight improvement of precision, the results of other four dataset decline, obviously the relationship depth between the terms can not exceed more than three, otherwise, it will add noise into meaning of original query. Actually, that is consistent with the optimal random walk steps in [28].

While the number of potentially active term-nodes in the network is large (our vocabulary size was around 6,000), the Markov transition matrices are very sparse: a typical matrix has about 900 non-zero entries. Limiting a walk to a maximum of 3steps typically results in less than 900 (15%) of potential nodes becoming active for title-length queries. Even so, using the network efficiently requires some planning. For example, the first time a word is seen, its score is cached since the aspect probabilities will not change over the lifetime of the query. The time to build the network can be reduced by doing off-line indexing for each article.

V. CONCLUSION

This paper presents a novel query Expansion method based on a Markov-based system to find related concepts and words. The documents in three datasets were thus expanded with related words, which were fed into a separate index. When combined with the regular index we report improvements over standard test collections using BM25 for those three datasets across several parameter settings, including default values, optimized parameters and parameters optimized in other datasets. In most of the cases the improvements are statistically significant, indicating that the information in the query expansion is useful. Similar to other expansion methods, parameter optimization has a stronger effect than our expansion strategy. The problem with parameter optimization is that in most real cases there is no tuning dataset.

The concept tree retrieval model based on Markov network, by introducing concept tree, and we make the entire query concept into account when select the expansion terms. This model is primarily concerned with two important problems of query expansion, namely with the selection and the weighting of expansion search terms.

The term selection relies on the dependence between the initial query terms and the similarity between terms of the collection. The weigh distribution adjusted according to a concept tree pruning algorithm, which remove the redundant information nodes, increase the weight of core query and core child node, and weaken the weight of non-core child node. The experiments carried out on the five test collections show that consistent improvement in the retrieval effectiveness can be expected.

In our future research we will concentrate on parameter tuning, the method involved in the proposed model to test to the optimal through the experiment, we will try to tune the parameters automatically, such as the EM algorithm. Furthermore, the parameters almost affected by the size of document size, so we want to test the approach in a larger test set, such as TREC.

ACKNOWLEDGMENT

Project supported by special fund of the Central University number: CHD2012JC087.

REFERENCES

[1] Manning, C. D., P. Raghavan, and H. Sch ¨utze. 2009. An introduction to information retrieval. *Cambridge University Press, UK*.

[2] Zhaoman Zhong, Cunhua Li, Yan Guan, Zongtian Liu, "A Method of Query Expansion Based on Event Ontology", *JCIT*, Vol. 7, No. 9, pp. 364 - 371, 2012.

[3] S. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at trec-3. In Online Proceedings of the Third Text Retrieval Conf, pp. 109–126, 1995.

[4] MandarMitra, AmitSinghal, ChrisBuckley. Improving automatic query expansion, *SIGIR*, pp. 206 - 214, 1998.

[5] Zuo jiali, Wang mingwen, Wang xi. Extended information retrieval model based on Markov network. *Journal of Tsinghua University (Science and Technology)*, 45(s1): pp. 1847-1852, 2005.

[6] Caoying, Wang mingwen, Tao hongliang. Information retrieval model based on Markov Network. *Journal of Shandong University (Natural Science)*, vol. 41, No. 3, pp. 126-130, 2006.

[7] H. E. Stiles, The association factor in information retrieval, *J. ACM*, 0004-5411 8:2, pp. 271-279, 1961

[8] K. van Rijsbergen. Information Retrieval (2nd edition). Butterworths, London, 1979.

[9] G. Salton and C. Buckley. On the use of spreading activation networks in automatic information retrieval. *SIGIR*, pp. 147-160, 1998.

[10] L M de Campos. Independency relationships and learning algorithm for singly connected networks. *J Exp Theory ArtifIntell*, 10(4): pp. 511-549. 1998.

[11] J. Lafferty and C. Zhai. Document languagemodels, query models, and risk minimization for information retrieval. *SIGIR*. pp. 111-119, 2001.

[12] K. Toutanova, C. Manning, and A. Y. Ng. Learning random walk models for inducing word dependency distributions. *Proc. of ICML* 2004.

[13] K. L. Kwok, L. Grunfeld, M. Chan. TREC-8 Ad-Hoc, Query and Filtering Track Experiments using PIRCS, *NIST special publication*, pp. 217-224, 2000.

[14] M. Quillian. Word concepts: a theory and simulation of some basic semantic capabilities. *Behav.* Sci. 12: pp. 410-430, 1967.

[15] C. C. Gottlieb, S. Kumar. Semantic clustering of index terms. *JACM* 15(4): pp. 493-513, 1968.

[16] K. Sparck Jones and E. O. Barber. What makes an automatic keyword classification effective? *JASIS* 22(3) 166-175, 1971.

[17] S. K. M. Wong and V. Raghavan. Vector space model of information retrieval-a re-evaluation. *SIGIR*, pp. 167-185, 1984.

[18] F. Crestani, Application of spreading activation techniques in information retrieval. *AI Review11* (6), pp. 453-482, 1997.

[19] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Proc. of 7th Int. WWW Conference*, pp. 107–117, 1998

[20] J. Kleinberg. Authoritative sources in a hyperlinked environment. *JACM*, 46:pp. 604–632, 1999

[21] J. Kleinberg and A. Tomkins. Applications of linear algebra in information retrieval and hypertext analysis.

© 2014 ACADEMY PUBLISHER

18th ACM Symp. on Principles of Database Systems, pp. 185-193, 1999.
[22] J. Xu and W. B. Croft. Query expansion using local and global analysis. *SIGIR* 1996. pp. 4-11
[23] E. M. Voorhees. Query expansion using lexical-semantic relations. *SIGIR* 1994. pp. 61-69.
[24] C. Shah and W. B. Croft. Evaluating high accuracy retrieval techniques. *SIGIR* 2004, pp. 2-9, 2004.
[25] A. J. Smola and R. Kondor. Kernels and regularization on graphs. Proc. of COLT 2003, Eds. B. Schölkopf and M. Warmuth, Lecture Notes in Computer Science, *Springer*.
[26] Sparck-Jones, K, Notes and references on early classification work. SIGIR Forum, 25(1), pp. 10-17, 1991.
[27] Yonggang Qiu, Hans-peter Frei. Concept based query expansion. *SIGIR*, pp. 160–169, 1993.
[28] Shisong, Wang mingwen, Tuwei, He shizhu. Extended information retrieval model based on the Markov network cliques. *Journal of Shandong University (Natural Science)*, vol. 46, No. 5, pp. 54-58, 2011.
[29] Makoto Iwayama, Atsushi Fujiii, NorikoKando, Yuzo Marukawa. An empirical study on Retrieval Models for Different Documet Genres: Patents and Newspaper Article. *SIGIR*, pp. 251-258, 2003.
[30] S. E. Robertson and D. A. Hull. The TREC-9 filtering track final report. *In TREC*, pp. 25–40, 2000.

© 2014 ACADEMY PUBLISHER