Passage retrieval using graph vertices comparison

Taoufiq Dkaki
IRIT-UMR5055
Université de Toulouse, France
taoufiq.dkaki@irit.fr

Josiane Mothe
IRIT-UMR5055
Université de Toulouse, France
josiane.mothe@irit.fr

Quoc Dinh Truong
IRIT-UMR5055
Université de Toulouse, France
truong@univ-tlse2.fr

Abstract

In this paper, we describe an Information retrieval Model based on graph comparison. It is inspired from previous work such as Kleinberg'Hits and Blondel et al.'s model. Unlike previous methods, our model considers different types of nodes: text nodes (elements to retrieve and query) and term nodes, so that the resulting graph is a bipartite graph. The results on passage retrieval task show that high precision is improved using this model.

1. Introduction

Information Retrieval Systems (IRS) usually consider documents as units of information: documents correspond to the units to be indexed and more importantly to the units to be retrieved.

The fact that most of the current systems choose this level of granularity is probably linked to the fact that the first automatic IRS handled short documents: generally secondary documents –references to the primary documents. This type of IRS intends to help the users to locate the relevant documents in the library: once secondary documents are selected the users can access paper versions manually. Nowadays, most of the systems handle electronic primary documents. As a consequence, the size of the documents has significantly increased, shifting from a few paragraphs to several pages or more. This change of the information nature should have lead to different retrieval mechanisms and interfaces. One of the cascading effects of this change is that a user may not be satisfied by a system that indicates a document is relevant without helping him to retrieve the specific parts in the documents that really answer his need.

In the literature, different approaches have been proposed to solve this problem by retrieving passages rather than documents. Some approaches aim at retrieving chunks or fixed-size windows of texts [1]; in these approaches the chunk dependency is taken into account by considering overlapping windows. Other approaches take into account the structure of documents and retrieve document components. First works considered SGML documents [2], [3]. More recently, the expansion of XML language leads to new research in component retrieval [4]; many approaches like [5] consider the nested composition of XML elements to decide which element should be retrieved. Passage retrieval in the context of non structured documents has recently been evaluated in the framework of the TREC program and more precisely in the Novelty Track [6]. In this track, a sub-task consists in retrieving relevant sentences from relevant documents. When considering non-structured documents; it is trickier to consider the dependency that exists between chunks of texts (in our case, sentences). In previous work [7], we acknowledge the sentence contextual dependency and we suggest a model that gives a higher score to sentences that follows sentences that got a high similarity score with the query.

In this paper, we present a graph-based model to represent the sentence dependencies. This model is a generalisation of the Hits algorithm [8], following the ideas developed in [9]. The similarity function we implement considers the structural similarities, which correspond to a relation-dependant function [10], [11]. We evaluate this model in the TREC framework. More precisely, in the context of relevance sub-task of the TREC novelty track and, considering precision at the top 5 documents (respectively 10 documents), we show that our model outperforms vector space model by 88% (respectively 131%).

The remaining of the paper is organized as follows. In section 2, we consider related works. Section 3 describes the background and starting points. Section 4 presents our model. Section 5 presents the evaluation and discusses the results. Finally, section 6 concludes this paper and indicates further research directions.

2. Related works

2.1. Passage retrieval on non-structured documents

TREC 2002 [6] defines passage retrieval at the sentence level in a general framework of redundancy detection. In the TREC framework, the sentences to be retrieved are selected from the documents that human evaluators had selected as relevant. Most of the systems used in TREC consider sentences as documents and applied their system modules at the sentence level instead of the document level without further changes. This, of course, does not take into account the contextual relationships that exist between sentences in a given document or paragraph. Individual sentences were indexed; then systems computed the similarity between sentences and queries. Blind relevance feedback was often used. [12] uses the traditional tf.idf and Language Modelling (LM) based models [13] combined with blind relevance feedback. [14] also uses tf.idf-based retrieval to measure the similarity between sentences and query; queries were expanded according to pseudo-relevance feedback. They studied different types of classifiers based on semantic and lexical features extracted from text analysis in order to remove possible non-relevant sentences. [15] chooses to expand the initial query adding both the terms that were semantically equivalent to the query terms and terms that co-occurred with the query terms. [16] combines blind relevance feedback and automatic sentence categorisation based on Support Vector Machine. [7] introduces a new approach based on term characteristics. Terms were categorised according to four classes: highly relevant, scarcely relevant, non-relevant (stop words), highly non-relevant.

2.2. Graph-based models

As we mentioned before, we propose a passage retrieval approach at the sentence level. We chose to model the sentence dependencies using graphs as they straightforwardly convey the idea of object relationships.

Many models consider network or graph based models such as inference networks [18], Bayesian belief network [19], Neural Network [20]. In this section we focus on the models that are most related to ours and that are variations of the PageRank or HITS models. Many works have considered the explicit inter-document relationships described through hyperlinks. PageRank [21], and HITS [8] are examples of such an approach. In these algorithms, citations and backlinks are used in addition to document content to decide the retrieved document order.

Other approaches, like ours, are based on documents that are not explicitly linked. In the context of document clustering, [22] models the document collection as a bipartite graph. Rather than clustering either terms or documents, using this model, documents and terms are clustered simultaneously. The method is based on graph partitioning. In the context of text summarization, [23] considers the importance of sentences by computing the centrality on graphs in which nodes correspond to sentences. The sentence similarity is based on cosine similarity which is used to define the graph representation of sentences. Also inspired from the PageRank principle, [24] consider a graph in which nodes are text units (a node can be either a term or a chunk of text) and compute the centrality of these nodes. The method considers undirected links between nodes; in addition these links are weighted. They applied their method to natural language processing applications such as key-phrase extraction and sentence extraction. [25] proposes a structural re-ranking method. The method is inspired by the PageRank and Hits principles, but, rather than using hyperlinks as an evidence of document relationships, these relationships are generated considering that the language model of a document content assigns high probability to another document content.

3. Motivation & Background

The main goal of graph based methods is to enhance the core IR-process of finding relevant documents in a collection of documents according to some user's need. More precisely, our method aims at enhancing both query-document matching and document ranking using a new graph-based similarity measure. This similarity measure we generate follows previous works related to graph matching and acknowledges the significance of structural similarity in making good comparisons as demonstrated in many psycho-cognitive studies. The method we propose is based on graph comparison and involves recursive computation of similarity.

3.1. Similarities in graphs

Many authors define similarity on two levels: surface and structural level. Surface similarity is defined as an attribute-oriented function while structural similarity is defined as a relation-dependent function. Figure 1 illustrates the different types of similarities [10, 11] which can be described as follows.

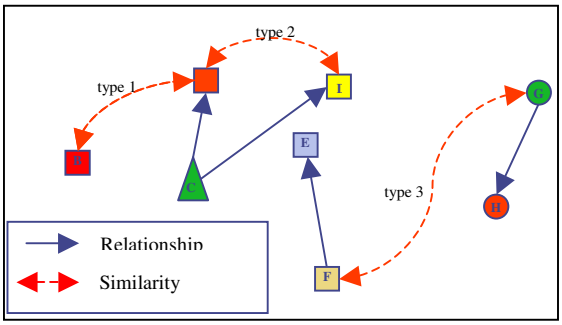

Figure 1. Structural and surface similarities between objects.

With regard to surface similarity (type 1 in figure 1), two objects are regarded as similar if they share the same characteristics (objects A and B in figure 1 are both red squares). There are two types of structural similarity. First-type of structural similarity also called "literal similarity" (type 2 figure 1) corresponds to the case when two objects are similar because they share the same characteristics (A and D in figure 1 are similar because they are both squares) and the same relation (they both have the same relation to *C*). Finally, second-type of structural similarity also called "analogy" (type 3 in figure 1) corresponds to the case when two objects are similar because they share the same role with regard to some relationship (F and G in the figure). Please read [10, 11] for more details.

In our approach, we consider the first type of structural similarity (type 2).

3.2. Blondel et al.'s model

[9] presents a graph-based model as a generalisation of the Hits algorithm [8]. Considering two oriented graphs, the similarity score s_{ij} between vertex *i* of a graph A and vertex *j* of a graph B can be computed as follows:

$$S_{ij} = \sum_{r:(r,i)\in E_B, t:(t,j)\in E_A} S_{rt} + \sum_{r:(i,r)\in E_B, t:(j,t)\in E_A} S_{rt} \quad (1)$$

E_A, E_B are the edge sets of graphs A and B.

In other words, given a node i from the graph B and a node j from the graph A, the similarity between i and j is computed, first, considering the sum of the similarities between each pair of nodes r and t, such as r belongs to the graph B and t belongs to the graph A and there is an edge that goes from r to i (incoming links), and second, considering the sum of the similarities between each pair of nodes r and t, such as there is an edge that relates i to r (outcoming links).

(1) can be rewritten and normalized in a matrix form as:

$$S_{k+1} = \frac{BS_kA^T + B^TS_kA}{\left\|BS_kA^T + B^TS_kA\right\|_F} \quad (2)$$

The computation of the similarity matrix is recursive; as a result, it is mandatory to define an initial similarity between nodes. Several initialisation strategies have been proposed in the literature [9, 17]. One of the most used strategies considers that the similarity between each pair at the first stage should be equal to 1, considering that there is no evidence that some nodes are closer than others and that each node should be considered equally. The issue of convergence of the similarity matrix series is discussed in [9].

This model could be applied as it is, as long as the handled objects are direct-linked. Hyperlinks provide such a graph structure. In our approach, we consider documents which are not explicitly linked. Contrary to other approaches that consider implicit inter-document relationships on the basis of the document similarities (e.g. cosine); in our approach, we consider implicit links on the more general basis of the terms they share –terms they are related to. As a result, node similarity is computed either between document nodes, between term nodes or between a document and a term node.

4. GVC model

The GVC (Graph Vertices Comparison) Model we propose is based on Blondel et al.'s model with regard to node similarity computation. However, it differs in several points that are described in the following sections.

4.1. Bipartite graph model

In IR we can consider three types of elements: units to retrieve (generally documents but in our approach they are passages), queries and terms. Terms are used either to represent documents or queries. Units to retrieve and queries are elements of the same nature with regard to the fact they can be (and generally are internally) represented by a set of terms. For that reason, the graph we consider is a bipartite one: nodes are of two types, terms or units of texts (documents or queries). In this model, there is no link between nodes of the same type; links relate units of texts and terms. Rather than considering two graphs, as it is generally the case when dealing with graph-vertices comparison, we consider a single graph and compute node similarities (the same graph is compared to itself).

4.2. Information retrieval principle

Information retrieval is based on the following process:

First we compute the similarity that exists between text nodes; following formula (2). This similarity is based on term node similarities (for those nodes that are in relationships with the considered texts), which are in turn computed according to text node similarities. These calculations are repeated until the system reaches convergence. Notice that a query is considered as a document is.

When the system converges, it results in a similarity square matrix in which columns and rows correspond to texts and terms. At this step, the query node row is extracted and sorted so that the most similar text nodes occur first. They correspond to the list of retrieved "documents".

4.3. Initial graph and similarities

Given N the number of text units to retrieve and M the number of indexing terms, the generated bipartite graph G is composed of N+M+1 nodes (1 corresponds to the query). Text nodes are not connected to each other, nor are term nodes. Terms correspond to indexing units; as a result, links between text units and terms are not directed. One could consider that if a term node i is connected to a text j; then the text j is connected to the term i. This would lead to computation cycles when considering the node similarities. For that reason, we arbitrary consider that only directed links going from document nodes to term nodes exist –links from terms to documents are not considered.

As explained in section 3.2, we also need to set the initial similarity between nodes. They have been set as follows:

The similarity between 2 text nodes is based on the cosine of the two texts (based on tf.idf, bag of words measure). In the same way, the similarity between 2 term nodes is initially computed as the cosine of the two terms (also based on tf.idf). Indeed, usually documents are represented as vectors in the term space; reversely, terms can be represented as vectors in the document space. Finally, the similarities between nodes of different nature are set to 0. That is to say similarities between any term and any document are set to 0.

4.4. Recursive similarities

Similarities between nodes are computed following the formula (2) with A=B=G.

$$S_{k+1} = \frac{GS_kG^T + G^TS_kG}{\left\| GS_kG^T + G^TS_kG \right\|_F}$$

Modulo the fact that we normalize the resulting S_{ij} so that S_{ii} =1 and S_{jj} =1.

[9] shows that the algorithm converges for the series 2k and 2k+1. That means that it is possible to stop the process once S_{ij} at the $(k+2)^{th}$ iteration do not differ (more than a threshold) from S_{ij} at the $(k)^{th}$ iteration and S_{ij} at the $(k+3)^{th}$ iteration do not differ from S_{ij} at the $(k+1)^{th}$ iteration. [17] shows that the adaptations that have been made to the graph model do not affect the convergence property. In addition, when considering the initial similarities as they have been set in section 4.3, [17] also show that whatever the iteration number is, the similarities between nodes of different nature remain equal to 0.

5. Evaluation

5.1. Collection

TREC Novelty 2004 is used to evaluate the model.

An example of a TREC topic is presented figure 3.

Topic: 35
Title: NATO, Poland, Czech Republic, Hungary
Type: event
Descriptive: Accession of new NATO members: Poland, Czech Republic, Hungary, in 1999.
Narrative: Identity of current and newly-invited members, statements of support for and opposition to NATO enlargement and steps in the accession process and related special events are relevant. Impact on the new members, i.e., requirements they must satisfy, and their expectations regarding the implications for them are relevant. Progress in the ratification process is relevant. Future plans for NATO expansion, identification of nations admitted on previous occasions, and comments on future NATO structure or strategy are not relevant.

Figure 2. TREC topic

A document set is associated to each topic. Each document is composed of sentences (provided by TREC), some are considered by TREC evaluators as relevant to the topic, other are considered as non relevant to the topic. The features of this collection are presented in figure 2.

#sentences per query	#relevant sent. per query	%relevant sent per query
1057	166	15,70

Figure 3. Features of the TREC Novelty track collection (averaged over queries)

5.2. Results

The first version of our model has been evaluated using the collection depicted in section 5.1.

Texts (sentences and queries) are processed using the following method:

- Stop words are removed,
- Remaining terms are stemmed
- Stems are weighted according to the tf.idf function

We get an average of 2877 term nodes and 1057 text nodes for per test collection. The graph has been built according to the method depicted in section 4 and similarities computed according to the formula in the same section. Notice that as similarities between documents and terms remain equal to 0 whatever the iteration, some elements of the similarity matrix do not need to be stored (we need to store the similarities between term nodes and between text nodes).

Results are compared to the vector space model: documents are represented by bags of weighted terms; the cosine measure is used to calculate the similarity between texts and the query and texts are retrieved according to decreasing similarity. This method corresponds to the baseline.

Figure 4 reports the recall precision graph obtained using the cosine measure and the GVC method.

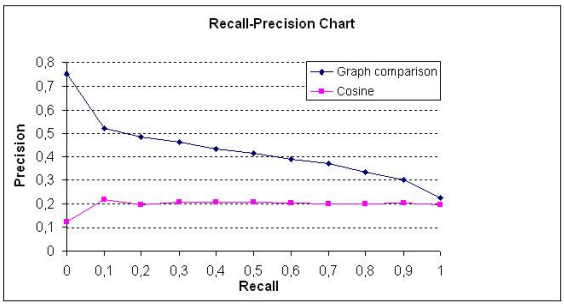

Figure 4. Recall-precision chart for TREC 2004

Because in our method, we did not set the number of retrieved elements, this number can vary from 1 to the entire collection. Figure 5 compares the results obtained when considering the baseline. It indicates the average precision according to the number of documents retrieved per topics. This figure shows that our method improves high precision (when a few documents are retrieved). The difference in terms of precision between the two methods decreases when the number of retrieved units becomes higher.

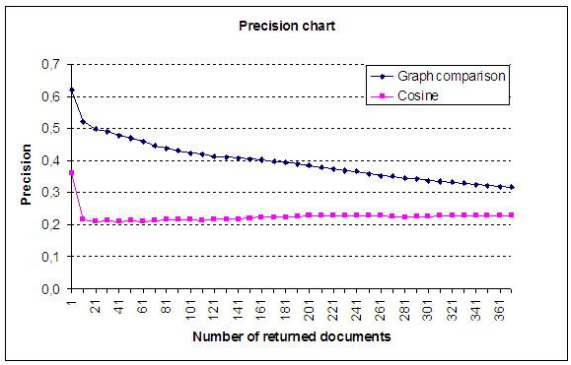

Figure 5. Average precision at top n for the TREC 50 topics

Finally, Figure 6 reports the precision at top documents (precision when 5 and 10 documents are retrieved) and the R-Precision (when R units are retrieved, R being the number of relevant units).

	Cosine	Graph	%
P@5	0.270	0.510	*+88.88%*
P@10	0.222	0.513	*+131.08%*
R-precision	0.237	0.400	*+68.78%*

Figure 6. Comparison with the vector space model

The details on the results obtained by the TREC participants are not available in detail. [26] reports that most of the participants that year get F-Measure between 0.36 and 0.40. The best participant got 0.42 as a F-Measure (LRIaze3 run).

6. Conclusions and future works

In this paper, we presented a new model based on graph comparison. Compare to the other model from which it is inspired, in our model, links do not need to be explicit. In addition, implicit links between text units or documents are based on explicit links between terms and documents. This model has been evaluated in the framework of passage retrieval (passages being defined as sentences). We have compared the model we defined with the vector space model and show high improvement, specifically for high precision, which is one of the most important criteria in the user's point of view. Further experiments have to be made on other collections and results must be compared to methods more sophisticated than the cosine.

The main weakness of our method is its high computational complexity which makes it unaffordable in the context of large document collections for now. However, we could use it in a MAC/FAC [27] architecture. MAC/FAC is a two-stage process in which a computationally cheap filter (MAC) is used to select a restricted subset of likely good candidates that are conveyed to a more accurate

and computationally expensive filtering process (FAC). Our graph vertices comparison method can be used as a FAC filter in association with a MAC method which will easily and quickly eliminate unnecessary documents. This is roughly what Kleinberg's HITS algorithm [8] does in order to reduce the computational its cost. HITS isolates a relatively small citation subgraph related to a given topic before detecting the authoritative 'sources' it contains.

In this preliminary model we consider that there is no relationship between terms; which is rather simplistic. Taxonomies or other terminological resources such as WordNet could be considered in order to better take into account term relationships. We also consider that documents are not linked each to other. When considering the web, this assumption is no more valid. In the context of the TREC collection we use, we could also consider that some links exist between the text units such as the fact that one sentence follows another sentence [7] or the fact two sentences belong to the same document or the same paragraph. We are working on a new model that would include all these types of links.

10. References

[1] Salton G., Allan J., Buckley C., Automatic structuring and retrieval of large text files, Communication de l'ACM, 37(2), 97-108, 1994.

[2] Wilkinson R., Effective retrieval of structured documents, Research and Development in Information Retrieval, SIGIR'94, 311-317, 1994.

[3] Corral M.-L., Mothe J., How to retrieve and display long structured documents ?, Proceedings of Basque International Workshop on Information Technology, BIWIT'95, 10-19, 1995.

[4] Initiative for the Evaluation of XML retrieval (http://qmir.dcs.qmw.ac.uk/INEX/).

[5] Hubert G., XML Retrieval Based on Direct Contribution of Query Components. In International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2005), Volume 3977:172-186, 2006.

[6] Harman D., Overview of the TREC 2002 novelty track, Text Retrieval Conference TREC 2002, 46-55, 2002.

[7] Dkaki T., Mothe J., Novelty track at IRIT-SIG, Proceedings of Text Retrieval Conference TREC 2004, 413-418, 2004.

[8] Kleinberg, J. M. Authoritative Sources in a Hyperlinked Environment. Journal of the ACM, 46(5):604-632, 1999.

[9] Blondel, V. D., Gajardo, A., Heymans, M., Senellart, P., and Van Dooren, P. A Measure of Similarity between Graph Vertices: Applications to Synonym Extraction and Web Searching. SIAM Rev. 46(4):647-666, 2004.

[10] Bracke, D. Vers un modèle théorique du transfert: les contraintes à respecter. Revue des sciences de l'éducation, XXIV(2) :235-266, 1998.

[11] Gentner, D., Ratterman, M. J., Forbus, K. D. The roles of similarity in transfer: Separating retrievability From inferential soundness. Cognitive Psychology, 25, 524-575, 1993.

[12] Larkey L.S., Allan J., Connell M.E., Bolivar A., Wade C., UMASS at TREC 2002: Cross Language and Novelty Tracks, Proceedings of Text Retrieval Conference TREC 2002, 721-732, 2002.

[13] J.M. Ponte, W.B. Croft, A language modelling approach to information retrieval, Proceedings of International ACM SIGIR conference on Research and Development in Information Retrieval, pp 275-281, 1998.

[14] Collins-Thompson K., Ogilvie P., Zhang Y., Callan J., Information filtering, Novelty detection, and named-page finding, Text Retrieval Conference TREC 2002, 107-118, 2002.

[15] Schiffman B., Experiments in Novelty Detection at Columbia University, Text Retrieval Conference TREC 2002, pp 188-196, 2002.

[16] Zhang M., Song R., Lin C., Ma S., Jiang Z., Jin Y., Liu Y., Zhao L., et Ma S., Expansion-based technologies in finding relevant and new information: THU TREC2002: Novelty Track Experiments, Proceedings of Text Retrieval Conference, pp 586-590, 2003.

[17] Dkaki T., Truong Q.D., Mothe J., Charrel P-J., A new method for information retrieval based on graph comparison, Proceedings of InSciT, 2006.

[18] Turtle H., Croft W.B., Evaluation of an inference network-based retrieval model, ACM Transactions on Information Systems (TOIS), Special issue on research and development in information retrieval, 9(3): 187 – 222, 1991.

[19] Ribeiro B. A. N., Richard Muntz, A belief network model for IR, Annual ACM Conference on Research and Development in Information Retrieval, Proceedings of international ACM SIGIR conference on Research and development in information retrieval, 253 – 260, 1996.

[20] Mothe J., Search mechanisms using a neural network-Comparison with the vector space model, 4th RIAO Intelligent Multimedia Information Retrieval Systems and Management, Vol.1, 275-294, 1994.

[21] Brin S. and Page L., The Anatomy of a Large-Scale Hypertextual Web Search Engine, http://infolab.stanford.edu/~backrub/google.html#ref

[22] Dhillon I.S., Co-clustering documents and words using bipartite spectral graph partitioning, Proceedings of

ACM SIGKDD international conference on Knowledge discovery and data mining, 269 - 274, 2001.

[23] Erkan G., Dragomir Radev D.R., LexRank: Graph-based Lexical Centrality as Salience in Text Summarization, Journal of Artificial Intelligence Research, 22:457-479, 2004.

[24] Mihalcea R. and Tarau P., Textrank: Bringing order into texts. In L. Dekang and W. Dekai, editors, Proceedings of EMNLP 2004, 404–411, 2004.

[25] Kurland O. and Lee L. PageRank without hyperlinks: Structural re-ranking using links induced by language models. Proceedings of International ACM SIGIR conference on Research and development in information retrieval, 306-313, 2005.

[26] Soboroff I., Overview of the TREC 2004 Novelty Track, Proceedings of Text Retrieval Conference TREC 2004, 2004.

[27] Forbus, K., Gentner, D. and Law, K. MAC/FAC: a model of similarity-based retrieval. Cognitive science (Cogn. sci.) ISSN 0364-0213 CODEN COGSD5, vol. 19, no2, pp. 141-205, 1995