Information Retrieval using a Macedonian Test Collection for Question Answering

Jasmina Armenska[1], Aleksandar Tomovski[2], Katerina Zdravkova[2], and Jovan Pehcevski[1]

[1] Faculty of Informatics, European University, Republic of Macedonia
{jasmina.armenska, jovan.pehcevski}@eurm.edu.mk
[2] Faculty of Natural Sciences and Mathematics, Ss. Cyril and Methodius University, Republic of Macedonia
aleksandar.tomovski@gmail.com, keti@ii.edu.mk

Abstract. Question answering systems solve many of the problems that users encounter when searching for focused information on the web and elsewhere. However, these systems cannot always adequately understand the user's question posed in a natural language, primarily because any particular language has its own specifics that have to be taken into account in the search process. When designing a system for answering questions posed in a natural language, there is a need of creating an appropriate test collection that will be used for testing the system's performance, as well as using an information retrieval method that will effectively answer questions for that collection. In this paper, we present a test collection we developed for answering questions in Macedonian language. We use this collection to test the performance of the vector space model with pivoted document length normalization. Preliminary experimental results show that our test collection can be effectively used to answer multiple-choice questions in Macedonian language.

Keywords: Information retrieval, Question answering, Macedonian test collection.

1 Introduction

The World Wide Web is an attractive resource for searching for valuable information. People increasingly rely on it to satisfy their daily information needs. However, despite the huge success of information retrieval systems in recent years, finding an answer to a particular question is not a simple task. First, it is quite difficult for the system to extract the query semantics and that of the underlying natural-language texts. Second, the amount of information on the Web increases every day, making the retrieval process even more difficult. Last, most of the users would not spend more time than necessary in finding the exact answer for their question. Indeed, even for a simple question users usually have to spend a lot of time, because the answer is rarely given in an explicit form. Therefore, they need to inspect every document retrieved by the system in order to find the required information.

As a result, more sophisticated retrieval tools for providing the correct answer to a user question are needed. This has led to the development of the so-called *question answering* systems (QA). QA systems provide the users with the answer for their question using certain information sources, such as the Web or a local document collection [10]. Two basic types of QA systems are distinguished: systems that try to answer the question by accessing structured information contained in a database; and systems that analyze unstructured information, such as plain text [4]. According to Belkin and Vickery [1], the systems of the first type are limited to a specific domain unlike the systems of the second type, which can cover different domains. A significant impact on question answering as a research area has been made by the Text REtrieval Conference (TREC), which has promoted a textual question answering track since 1999 [3]. Mulder was developed as a result, which is the first general-purpose, fully-automated question answering system available on the Web [4].

Currently, there are several textual question answering systems that include different techniques and architectures. Most of them have a number of components in common, and these are: question analysis, retrieval of relevant documents, document analysis and answer selection [5]. The question analysis component includes morpho-syntactic analysis of the given user question posed in a natural language text, as well as determining its type and consequently the type of the answer that is expected. Depending on the performed morpho-syntactic analysis, a retrieval query is formulated in order to identify relevant documents that are likely to contain the answer of the original question. Document analysis component extracts a number of candidate answers that are then ranked by the answer selection module according to their estimated likelihood of relevance.

Test collections are usually used for measuring the performances of information retrieval systems. A certain test collection consists of three parts: *a document collection* that comprises documents written in a particular language (in our case in Macedonian language), a set of *user queries* required for information retrieval from the document collection (in our case questions), and a set of *relevant documents* that correspond to the user queries (in our case relevant answers). Well-known forums for creating test collections for the most popular world languages are: Text REtrieval Conference (TREC)[1], INitiative for the Evaluation of XML retrieval (INEX)[2], NII Test Collection for IR Systems (NTCIR)[3] and Cross-Language Evaluation Forum (CLEF)[4]. They are primarily maintained for testing the well-established (or new) models for information retrieval on different test collections and for exchange of useful information and knowledge from the gained experiences. To the best of our knowledge, there is no existing Macedonian test collection that can be used for empirical testing of various question answering retrieval methods.

In this paper, we describe a test collection that consists of documents and questions posed in Macedonian language, as well as their relevant answers. We use this test collection to investigate the effectiveness of one of the best performing information retrieval methods, the pivoted cosine document length normalization [8].

---

[1] http://trec.nist.gov/

[2] http://www.inex.otago.ac.nz/

[3] http://research.nii.ac.jp/ntcir/index-en.html

[4] http://www.clef-campaign.org/

2 A Macedonian Test Collection for Question Answering

We have created our own test collection that can be used for developing and testing systems for answering questions posed in Macedonian language. The collection consists of four documents and 163 multiple-choice questions taken from the courses History of Informatics and Computer Applications that are part of the curriculum of the Institute of Informatics at the Faculty of Natural Sciences and Mathematics at the Ss. Cyril and Methodius University in Skopje. The document names are: "A brief history of computers", "Introductory concepts", "Hardware" and "Software". Fig. 1 shows a snippet taken from the document "Hardware".

Двата најзастапени влезни уреда на сметачите се тастатурата (keyboard) и глувчето (mouse). Тие се најчесто поврзани со кутијата на сметачот со помош на кабел, но можат да бидат на далечинско управување (remote control) или безжични (wireless).

Fig. 1. A snippet from the document "Hardware".

All the questions are extracted from these four documents, and every question belongs to only one of the existing question types: Who, When, Why, What, What (description), What (size), How, How Many, Where and Other (for uncategorized questions). Four answers for every question are given and only one of them is correct.

Below is an example of a question whose answer can be found in the document "Hardware", which belongs to the Who question type. There are four answers given for the question, and only the first one is correct.

Кои се најважните влезни уреди?

1. глувчето и тастатурата (correct)
2. екранот и тастатурата
3. глувчето и екранот
4. екранот и мониторот

Our test collection consists of a set of 163 questions, divided into two subsets: a training set (containing 83 questions) and a testing set (containing the remaining 80 questions). The training set is used to determine the optimal values of the tuning parameters in our retrieval system. That is, we set optimal values for those parameters that maximize the retrieval performance on the training set, and then use these optimal parameter values on the testing set with the assumption that for these values our system will produce the best results. The testing set is therefore used to confirm the retrieval performance previously achieved on the training set.

2.1 Training Set

Table 1 shows the overall breakdown of questions comprising the training set, i.e. their distribution over documents (columns) and over question types (rows). We observe that most of the questions belong to the document "Hardware" (42%), followed by "A brief history of the computers" (29%), "Introductory concepts" (17%)

and “Software” (12%). On the other hand, the two mostly used question types are What (43%) and Who (28%), with the rest of the question types almost uniformly distributed.

Table 1. A breakdown of questions comprising the training set.

Question Type/Document	A brief history of computers	Introductory concepts	Hardware	Software	Total
Who	11	2	7	3	**23**
What	7	6	18	5	**36**
When	3	/	/	/	**3**
Why	1	/	1	/	**2**
What (desc)	1	1	2	/	**4**
What (size)	1	/	/	/	**1**
How	/	1	2	1	**4**
How Many	/	1	2	/	**3**
Where	/	/	3	1	**4**
Other	/	3	/	/	**3**
Total	**24**	**14**	**35**	**10**	**83**

2.2 Testing Set

Table 2 shows the overall breakdown of questions comprising the testing set, i.e. their distribution over documents (columns) and over question types (rows). It can be noticed that very similar distribution of questions is observed for the testing set as it was previously observed for the training set.

Table 2. A breakdown of questions comprising the testing set.

Question Type/Document	A brief history of computers	Introductory concepts	Hardware	Software	Total
Who	11	/	3	2	**16**
What	8	6	21	7	**42**
When	3	/	/	/	**3**
Why	/	/	/	/	**0**
What (desc)	1	/	1	/	**2**
What (size)	/	/	1	/	**1**
How	/	2	3	1	**6**
How Many	/	1	2	/	**3**
Where	/	/	2	/	**2**
Other	/	5	/	/	**5**
Total	**23**	**14**	**33**	**10**	**80**

3 Information Retrieval Methods

In order for the information retrieval (IR) system to understand the user’s need, it has to be represented by a query request comprising a set of terms. Most of the

existing IR systems index terms, phrases or other document content identification units [6]. Usually the indexing is based on a structure called *inverted index*, which is considered as one of the most efficient ways to process vast amounts of text [11].

3.1 Collection Statistics

The similarity of a document to a given query indicates how *closely* the content of the document matches the content of the query. In order to measure this similarity, statistical information about the distribution of the query terms in the document as well as in the whole collection is needed. Many similarity measures have been proposed, and most of them implement one of the three major IR models: the vector space model [7], the probabilistic model [9], and the language model [12].

We define the following statistics for a given document collection and a query:

- q - a query
- t - a query term
- d - a document
- N - number of documents in the collection
- $tf_{t,d}$ - the frequency of t in document d
- df_t - number of documents containing the term t (document frequency)
- $tf_{t,q}$ - the frequency of t in query q
- V - number of unique terms from the document collection

3.2 Vector Space Model

In the statistically based vector space model, each document is represented by a vector, whose components are the unique terms derived from the documents in the collection, with associated weights representing the importance of the terms in the document as well as in the whole document collection. The set of documents in the collection may thus be viewed as a set of vectors in a vector space. On the other hand, each query can also be treated as a short document so that it too can be represented by a vector in the same vector space as that used for the documents in the collection.

Three important components that affect the term weight in a given collection are the term frequency (tf), the inverse document frequency (idf) and the document length. The weight of a term in a document vector can be determined in many ways. A common approach uses the so-called $tf \times idf$ method, in which the weight of a term is determined by using only these two factors [7].

More formally, one representation of the weight of the term t in a document d is:

$$w_{t,d} = tf_{t,d} \times idf_t = tf_{t,d} \times log \frac{N}{df_t} \quad (1)$$

The standard way of measuring the similarity between a document d and a query q is the cosine measure, which determines the angle between their vector representations $\vec{V}_d$ and $\vec{V}_q$ in the V - dimensional Euclidean space:

$$sim(q,d) = \frac{\vec{V}_d \cdot \vec{V}_q}{V_d \cdot V_q} = \frac{\sum_{t \in V} w_{t,d} \times w_{t,q}}{\sqrt{\sum_{t \in V} w_{t,d}^2} \times \sqrt{\sum_{t \in V} w_{t,q}^2}} \quad (2)$$

The denominator in this equation is a product of the Euclidean lengths of the vectors $\vec{V}_d$ and $\vec{V}_q$ which represent their *cosine document length normalizations*.

The document length normalization is a way to penalize the term weights for a document in accordance with its length. Empirical results obtained from TREC documents and queries show that more effective are those techniques which implement normalization strategies that retrieve documents with similar chances to their probability of relevance [8]. Cosine normalization has a tendency to retrieve short documents with higher probability than their probability of relevance. On the other hand, the probability to retrieve longer documents is lower than their probability of relevance. In order to promote retrieval of longer documents and at the same time to retrieve less short documents, *pivoted document length normalization* is used.

3.3 Pivoted Document Length Normalization

The main idea is that the probability of document retrieval is inversely related to the normalization factor. It means that increasing the chances of retrieval of longer documents can be achieved by lowering the value of the normalization factor for those documents, and vice-versa. If the curves of probability of retrieval and probability of relevance are plotted against the document length, they intersect in a point called *pivot*. Usually, the documents on one side of the pivot are retrieved with probability higher than their probability of relevance and the documents on the other side of the pivot are retrieved with probability lower than their probability of relevance. The idea of the pivot normalization is to rotate the curve of probability of retrieval counter-clockwise around the pivot so that it more closely matches the curve of probability of relevance [8].

The simplest implementation of the pivoted cosine normalization is achieved by using a normalization factor that is linear in vector length:

$$u = (1 - S) + S \cdot \frac{V_d}{V_{avg}} \quad (3)$$

Here, S is a *slope* that receives values in the interval [0, 1], and V_{avg} represents an average length of the documents in the collection. This normalization factor shows that the most appropriate length has a document with average length, which means that the weights of its terms should remain unchanged. It should be emphasized that the cosine normalization is a specific case of the pivoted normalization (S=1).

Initial tests show that the deviation of the retrieval probability from the probability of relevance is systematic across different query sets and different documents collections, where an optimal slope value S=0.2 is identified [8]. This suggests that the slope trained on one collection can effectively be used on another one. However, recent research shows that the slope should be carefully calibrated according to the document collection [2]. In our experimental results, we will also experiment with different values for the slope parameter S on the training set in order to determine its optimal value that can be achieved on our Macedonian test collection.

4 Experiments and results

In this section we present results from the practical implementation of the vector space model with pivoted document length normalization, applied to our Macedonian test collection for question answering. The implemented retrieval system is developed in C# (.Net Framework 3.5). Matrices and vectors are used as basic data structures that are manipulated in many ways in order to get the results.

4.1 Retrieval strategies

Two phases are used to find answers for questions posed in Macedonian language.

Phase 1 (Document selection)

In this phase, one of the four documents that is most likely to contain the correct answer to a question is first selected. Two types of queries are used by our system to select the right document: the first query contains only the question, while the second query contains the question combined with all of the provided answers. Regardless of the query type, the highest ranked document is considered to contain the correct answer to the question, and is further processed in phase two.

Phase 2 (Answering the question)

Based on the selected document in phase one, our retrieval system utilizes one of the following two strategies to select the correct answer for the question: (1) using the whole document, or (2) using document passages. The passages are identified by the way MS Word defines paragraphs – namely, each section that ends with pressed Enter (new line) is treated as a retrieval passage.

When using the whole document as a retrieval strategy, the system uses four queries, each containing the initial question *combined* with only one of the four

provided answers. In this case, a correct answer is considered to be the answer for which the corresponding query returns the highest ranking score for the document.

When using document passages as a retrieval strategy, the system again uses the same four queries, only this time for each query the score of the highest ranking passage is first noted. The four scores (obtained for each of the four queries) are then sorted in a descending order, and a correct answer is considered to be the answer for which the corresponding query returns the highest score for the top ranked passage.

In both cases, the score for a given document or passage is obtained by using the vector space model with pivoted document length normalization.

4.2 Training Set

We now present experimental results obtained on the training set of our test collection, when using the vector space model with pivoted normalization. The idea is to determine the values for the parameters that maximize the retrieval performance.

1) Phase 1: Selecting the right document

In this phase, we want to determine which of the two query types is better for selecting the right document that contains the answer for a given question. For this experiment we use the vector space model with cosine normalization (S=1). We have found that using the question combined with all of the answers as a query produces 87% accuracy (across all the 83 questions in the training set), as opposed to when using the question alone that produces 72%. In the further analysis we therefore use the question combined with all of the answers as a query to our system.

In order to determine the optimal value for the slope parameter S when selecting the document that contains the answer to a particular question, we analyzed twenty values for the slope S, in the range between 0 and 1, with a step of 0.05. We found that there are two values for S (0.25 and 0.30) for which an accuracy of 92% is obtained (percent of questions with correctly selected documents). This is a 5% relative performance improvement against the previous value obtained by using the vector space model with cosine normalization.

2) Phase 2: Finding the correct answer

In this phase, we want to determine which of the two retrieval strategies works better for selecting the correct answer for a given question. We have found that, when using the whole document (previously selected in phase one), there is a 35% accuracy obtained by our system (which represents the percent of correctly answered questions by the system). Since only one document is used for answering the questions, any value of S in this case produces the same result.

Fig. 2 shows the accuracy obtained by our system when using document passages as a retrieval strategy, as the slope parameter S varies between 0 and 1. We observe that there are three values for S (0.50, 0.80, and 0.85) for which an accuracy of 56% is obtained. This is a 62% relative performance improvement against the previous value obtained when using the whole document as a retrieval strategy.

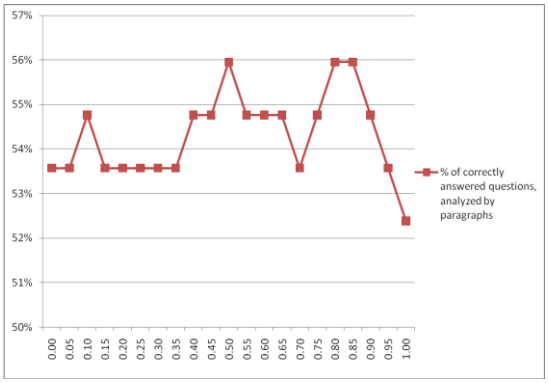

Fig. 2. Percent of correctly answered questions when using passages from previously selected document on the training set, as the slope parameter S varies between 0 and 1.

4.3 Testing Set

We now present experimental results obtained on the testing set of our test collection, when using the vector space model with pivoted normalization. The idea is to confirm the performances for the optimal values of retrieval parameters, obtained previously on the training set.

1) Phase 1: Selecting the right document

In this phase, we compared two retrieval methods for selecting the right document on the testing set: one that uses the vector space model with cosine normalization (S=1) as a baseline, and another using the optimal value for the slope parameter S, previously determined on the training set (S=0.3). With the optimal S value we have achieved 95% accuracy in selecting the right document, against the baseline where we achieved 87% accuracy (which is around 8% relative performance improvement).

2) Phase 2: Finding the correct answer

When using the whole document as a retrieval strategy on the testing set, our system obtained an accuracy of 34%, which is almost identical to the accuracy obtained on the training set (35%).

When using document passages as a retrieval strategy, we compared two retrieval methods for finding the correct answer on the testing set: one that uses the vector space model with cosine normalization (S=1) as a baseline, for which we obtained 51% accuracy; and another using the optimal value for the slope parameter S, previously determined on the training set (S=0.85), for which we obtained 55% accuracy (a 7% relative performance improvement).

5 Conclusion and future work

The technology for answering questions is very important part of (focused) information retrieval, because the precise question answering is the key in handling the information explosion. The main goal with the research presented in this paper was creating a test collection for question answering in Macedonian language, in order to implement and test the well-established IR methods for question answering purposes. Our experiments with the vector space model using pivoted document length normalization show that the document (or passage) lengths, as well as the choice of a retrieval strategy from the document itself are key factors in determining the correct answer to a particular question.

In the future, we intend to enrich the Macedonian test collection with additional documents and questions, as well as to share this collection with existing forums for research purposes, in order to improve the performances on other existing retrieval methods. We also plan to compare the performances of other well-established (or novel) IR methods on the Macedonian test collection.

References

1. N.J. Belkin and A. Vickery. *Interaction in information systems*. The British Library, 1985.
2. A. Chowdhury, M. Catherine McCabe, D. Grossman and O. Frieder. Document normalization revisited. In *Proceedings of the ACM-SIGIR International Conference on Research and Development in Information Retrieval*, pages 381–382, Tampere, Finland, 2002.
3. H.T. Dang, D. Kelly, and J. Lin. Overview of the TREC 2007 Question Answering Track. In *NIST Special Publication 500-274: The Sixteenth Text REtrieval Conference Proceedings (TREC 2007)*, Gaithersburg, Maryland, 2007.
4. C. Kwok, O. Etzioni, D. Weld. Scaling Question Answering to the Web. *ACM Transactions on Information Systems*, 19(3):242–262, 2001.
5. B. Magnini, M. Negri, R. Prevete, H. Tanev. Mining the Web to validate answers to natural language questions. In *Proceedings of Data Mining 2002* , Bologna, Italy, 2002.
6. C. Manning, P. Raghavan, H. Schutze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
7. G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.
8. A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *Proceedings of the ACM-SIGIR International Conference on Research and Development in Information Retrieval*, pages 21–29, Zurich, Switzerland, 1996.
9. K. Sparck Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: Development and comparative experiments. Parts 1 and 2. *Information Processing and Management*, 36(6):779–840, 2000.
10. M. Sultan, *Multiple Choice Question Answering*, MSc thesis, University of Sheffield, 2006.
11. I. H. Witten, A. Moffat, and T. C. Bell. *Managing Gigabytes: Compressing and Indexing Documents and Images, Second Edition*. Morgan Kaufmann Publishers, 1999.
12. C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.