

Improving Web Search Result using Semantic Similarity with Re-ranking

¹Rutuja Ajmire, ²A. V. Deorankar, ³Dr. P. N. Chatur

^{1,2,3}Dept. of CSE, Government College of Engineering, Amravati, Maharashtra, India

Abstract

The Aim of this paper is to re-ranking the web search result using semantic similarity to improve the quality of search engines. First obtain top N results returned by search engine such as Google, and then use semantic similarities between the Content obtain from the web search result and the users query. A semantic similarity algorithm based on WordNet ontology which is used to calculate the similarity of each snippet to each of the return result. And then based on the similarity re-ranking is performed. A balanced similarity ranking method combined with Google's rank. Here first we convert the ranking position to an importance score for semantics instead of keyword matching which can better adapt timeliness of the pages is used to rank these Web pages.

Keywords

Search Engine, Information Retrieval, Re-Ranking, Semantic Similarity

I. Introduction

The Aim of this paper is to propose re-ranking method which employs semantic similarity to improve the quality of search results. As the information on the Internet explodes astonishingly, search engines play a more and more important role. However, due to the diversity of web users' search requests; it is necessary for search engines to improve their keyword-based search techniques

Research on search engine has a long history in philosophy, psychology, and artificial intelligence, and various perspectives have been suggested from both academic field and industry. Users can obtain different kinds of information associated with their keywords from current commercial search engines, such as Google. However, the results of these search engines are lack of semantic relationship among each other since they may be ordered by importance only. Furthermore, there are many information redundancies of these results, and it is always time-consuming for users to find out the most relevant information they want. Hence, this paper introduce a ranking method based on the calculation of semantic similarity of web search results to improve the user experience of using search engines, especially finding out the most relevant information they want.

Whenever Re-ranking algorithm is performed two important things should be taken into account the first is the original queries and the second is the initial retrieval scores. A web search engine is designed to search for information on the World on the World Wide Web and FTP servers. Google Search is the most-used search engine on the World Wide Web receiving several hundred million queries each day through its various services. Snippets, a brief window of text extracted by a search engine around the query term in a document, provide useful information regarding the local context of the query term. [3]

Semantic search seeks to improve search accuracy by understanding searcher intent and the contextual meaning of terms as they appear in the searchable data space, whether on the Web or within a closed system, to generate more relevant results. Hence, this paper proposes an automatic ranking system based on the calculation of semantic similarity of these results to improve the user experience.

This paper is based on the semantic similarity. For calculating the semantic similarity which is based on WordNet each snippet fetched from the N top result of the search engine. A balanced similarity ranking method combined with Google's rank and timeliness of the pages is proposed to rank these snippets. And hence Re-ranking is done based on the similarity.

II. Literature Survey

A. Related Work

Keyword matching techniques is used by the current search engine which has following Weaknesses. First, many times Web users cannot express their search intention accurately using several keywords. And hence the exactly-matched results do not consequentially satisfy the web users. Secondly, keyword matching cannot guarantee the selected candidates have high correlation with the user query, given the different positions and meanings of the keywords. Third, under the circumstances of keyword matching, the top ranked search results for a given query must contain the keywords as much as possible; otherwise they will lose their ranking positions although their contents exactly discuss the same thing. This will also lead to an awkward situation: spammers try their best to pollute the web document corpus with term spamming tricks such as repetition, dumping and weaving.

Another problem about current search engines is their ranking schemes. PageRank is the most popular ranking algorithm; however, it is based on the popularity of web documents, not the quality. Therefore, a newborn web document usually cannot get highly-ranked positions due to their freshness and thus little reputation. The search based on lexical semantics in which content are semantically check instead of keyword matching which can better adapt to the thinking pattern of human beings, and thus search results are more relevant to users' search intention. Meanwhile, using semantic factors can conciliate the freshness and make the high-relevant new pages get moderate rank promotion. In our work, we fetch the top N results returned by search engines such as Google for user queries, and use semantic similarities between the candidate and the query to re-rank the results. We first convert the ranking position to an importance score for each candidate. Then we combine the semantic similarity score with this initial importance score and finally we get the new ranks..[1]

In this paper we are interested specifically in the semantic evaluation of the results returned by search engines. This choice is motivated by their popularity in the Web community on the one hand and the degree of selectivity that they offer on the other. More precisely, this system allows to: [2]

- Retrieve the results returned by search engines
- Check the information content of each returned page.
- Project the user query on the linguistic resource, the WordNet ontology.
- Measure the results relevance by calculating the relevance degree of each of them.
- Generate a semantic rank of results according to the calculated relevance based on their degree of informativeness.

- Assign a score to each web page based on its position in the new ranking.

This system is based partly on a linguistic resource (WordNet ontology) for the query semantic projection and on the other hand, a calculation model for measuring the relevance 'document/query'. In the following we are justifying our choices in terms of the chosen linguistic resource and the used IR model.

In literature survey Graph based ranking algorithms [6] was develop which can be very helpful when searching for important pages in the World Wide Web. Such algorithms capitalize on the existence of explicit links (e.g., hyperlinks, citations) between the graph vertices. In the case of flat text collections, neither links nor citations exist, so the need to devise implicit edges between text keywords or sentences arises. One solution is to exploit the contextual information of terms and create semantic graphs from text based on content similarity.

Semantic Rank, a new algorithm for ranking text segments. Semantic Rank comprises two steps:

- Creation of semantic graphs from text using both semantic and statistical information,
- Application of a graph-based ranking algorithm that exploits the edges' weights.

This work addresses the problem of extracting the most representative keywords and sentences from text as a means of text summarization. More specifically, Semantic Rank capitalizes on the creation of term and sentence graphs from text and on graph based ranking algorithms in order to support the following tasks:

- Keyword extraction from text, which is performed by selecting the top-ranked terms as the most representative ones,
- Text summarization, which is done by selecting the top ranked sentences as the most representative ones.

For this reason, a survey of research works in keyword extraction and text summarization, with emphasis on graph-based approaches is necessary to understand the requirements for these tasks, to locate benchmark data sets, and state of the art graph-based approaches for the comparative evaluation.

B. WordNet

WordNet is a lexical database for the English language. It groups English words into sets of synonyms called synsets, provides short, general definitions, and records the various semantic relations between these synonym sets. The purpose is twofold: to produce a combination of dictionary and thesaurus that is more intuitively usable, and to support automatic text analysis and artificial intelligence applications. WordNet distinguishes between nouns, verbs, adjectives and adverbs because they follow different grammatical rules. It does not include prepositions, determiners etc. Every synset contains a group of synonymous words or collocations.[8]

While semantic relations apply to all members of a synset because they share a meaning but are all mutually synonyms, words can also be connected to other words through lexical relations, including antonyms (opposites of each other) which are derivationally related, as well.

WordNet also provides the polysemy count of a word: the number of synsets that contain the word. If a word participates in several synsets (i.e. has several senses) then typically some senses are much more common than others. WordNet quantifies this by the frequency score: in which several sample texts have all words semantically tagged with the corresponding synset, and then a count provided indicating how often a word appears in a specific

sense.

The morphology functions of the software distributed with the database try to deduce the lemma or root form of a word from the user's input; only the root form is stored in the database unless it has irregular inflected forms.

WordNet can be interpreted and used as a lexical ontology in the computer science sense. However, such ontology should normally be corrected before being used since it contains hundreds of basic semantic inconsistencies such as (i) the existence of common specializations for exclusive categories and (ii) redundancies in the specialization hierarchy. Furthermore, transforming WordNet into a lexical ontology usable for knowledge representation should normally also involve (i) distinguishing the specialization relations into subtype of and instance of relations, and (ii) associating intuitive unique identifiers to each category

C. Semantic Similarity

For Calculating semantic similarity, ontology must be specified first. For example, WordNet1 is a very famous and widely used ontology. Based on WordNet, researchers have already put forward some semantic similarity formulas. For example, Leacock and Chodorow propose the following formula to compute the semantic between two concepts [7]:

$$sim(\pi_1, \pi_2) = -\log \frac{len(\pi_1, \pi_2)}{2 * \max_{\pi \in wordnet} depth(\pi)} \quad (1)$$

where $len(\pi_1, \pi_2)$ is the length of the shortest path between concept π_1 and π_2 in WordNet and $depth(_)$ is the length of the path from to the root.

D. Ranking Snippets by Semantic Similarity

In this section, systematic method is proposed to classify the results associated with keywords from Google into corresponding categories by comparing their semantic similarity automatically. Google results associated with corresponding keywords are used as the input for this system.

A WordNet based semantic similarity algorithm is proposed to calculate the similarity of these extracted words. A combined similarity for each snippet is calculated from the similarity of the word, which is the element of one snippet. This combined similarity value is used as a factor for ranking of snippets based on similarity. In each ranking, the snippets are ordered in a descending sequence according to the similarity and other factors.

III. Project Design

A. Overall Design of Project Work Flow

The project is completely designed in two different Phases as shown in the fig. 1.

As Shown in fig. 1, first a query is fire on a backend search engine like Google, Google will return a result which is based on keyword matching. For improving the Web search result this paper present a new approach. First snippets are fetched from the N top result return by the Google search engine. Stemming and part of speech are performed on the snippets and then using ontology like WordNet a semantic similarity is calculated. Based on the Semantic Similarity of the snippet and the query and the importance score of the original ranking a new re-ranking algorithm is applied which will improve the quality of search engine.

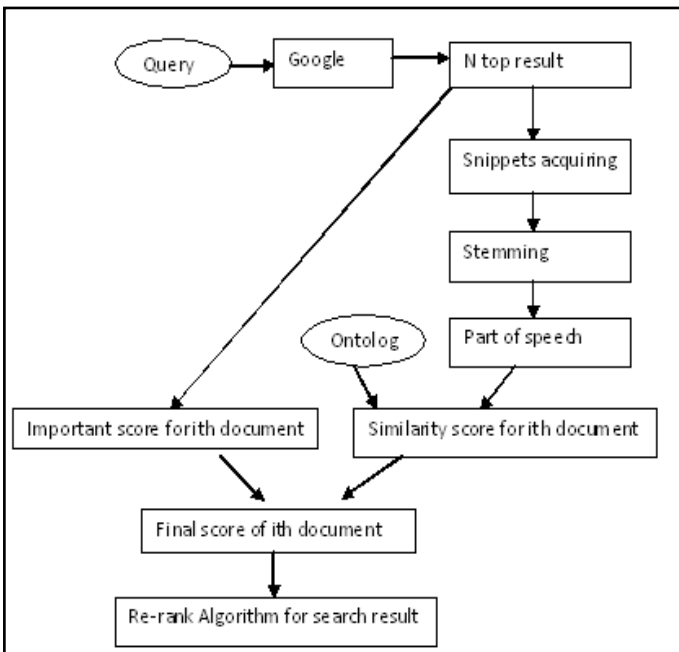


Fig. 1: Re-ranking Method

The main stages of the project are as follows:-

- Pre-processing on retrieved information from search engine.
- Calculating semantic similarity score between snippets and topics.
- Calculating the importance Score for each document.
- Balanced method of ranking web pages from semantic similarity and importance score.
- Semantic Ranking.

1. Pre-Processing on Retrieved Information from Search Engine

The first phase is to retrieve the snippets from Google results. From each paragraph, system can extract one snippet in free text by eliminating the HTML tags. A snippet is defined as: Snippet = paragraph (Title) + paragraph (Abstraction).

First remove stop words by comparing the words in each snippet with the one in a stop list. The stemming process and POS are employed to disambiguate the forms and the appropriate meaning of the words in each snippet. The system utilizes TreeTagger, which is a language independent POS tagger, to implement these tasks. The final outcome of this phase is a group of “processed snippets”, which consists of a set of extracted words.

2. Calculating Semantic Similarity Between Snippets and Query

In order to re-rank the snippets into the most relevant user’s query, an algorithm is proposed to calculate the similarity between the snippets and the topics. In case of traditional semantic similarity calculation method only word distance is calculated but in this paper presents two important factors. The first factor is based on the path of concepts in WordNet. The second one is based on the information content of concepts which comes from entropy measurement. Moreover, a coefficient is employed to balance the weight of the two factors for accuracy revision.

We utilize both the logarithm of the concept of distance, which refers to the number of the nodes between two words, to reflect the change trend of the similarity and the density of two words on the semantic path. The corresponding equation based on these

for term T_1 and T_2 is as follows:

$$sim_{path}(T_1, T_2) = - \frac{1}{\left(\frac{depth(T_1) + depth(T_2)}{2Max_{depth}} \right) * \left(\log \frac{1}{Distance(T_1, T_2) + 1} \right)} \quad (2)$$

where $Depth(T)$ is the number of levels of the term counting from the root node “Entity”; the Max_{depth} is the maximum number of the level in the taxonomy of WordNet; $Distance(T_1, T_2)$ is the number of common hyponyms nodes between the two terms including themselves, in which the nodes are the closest common parent nodes in WordNet.

3. Calculating The Importance Score for each Document

This paper is heavily depended on the search engine’s quality and result, how to grade the web pages returned from the search engine is important. So we need to find criteria to measure each web page. Before introducing our method, we would like to show one important concept, the DCG, short for Discounted Cumulative Goal. It is a common way to evaluate the search result quality using DCG. The expression (3) of DCG is as follows.

$$DCG_p = \sum_{i=1}^p \frac{2^{rel_i} - 1}{\log_2(i + 1)} \quad (3)$$

Where p is PageRank serial number and rel_i is the graded relevance of the result at position i . This expression has a discount factor, which makes the lower the rank; the smaller share of the document value is added to the cumulated gain. The lower the ranked position of a relevant document - of any relevance level - the less valuable it is for the user, because it is less likely that the user will examine the document due to time, effort, and cumulated information from documents that already seen.

Let us go back to how to measure the importance of the results at different positions. As we know, the search results are returned by search engines according to their importance and relevance. The most important web pages usually are returned at the top positions, and hence attract much more attention from users. On the contrary, the unimportant pages are returned at the bottom positions. Therefore, a discount factor is needed which progressively reduces the document value as its rank decreases. An intuitive and simple way for this requirement is also to use a log value as the divisor, just like the DCG does. We propose the following formula to calculate each web’s importance score.

$$importance(i) = \frac{1 - (i - 1)/tot}{\log_2(i + 1)} \quad (4)$$

Where i is the original PageRank serial number (i.e., the original ranking position) and tot is the number of the fetched web pages for a query. The formula indicates that the top results have significant importance to the search keywords and thereby are much valuable for web users. In order to make calculation simple, the result is normalized. For example, if i equals to 1, the importance of the first web page is 1, which is the highest score. Along with the increasing of the ranking position number of a returned result, its importance score is decreased sharply

4. Semantic Ranking

A Semantic Rank algorithm is proposed for re-ranking the web search result which is based on the importance score of Original ranking and the semantic similarity.

IV. Conclusion

In this paper we propose a new method to improve the quality of web search engine. This method first obtain the result from the engine, fetch the N top result and calculate the semantic similarity between Query and the snippets fetch from the search engine. And the Semantic Re-ranking algorithm is proposed which will re-rank the original search result based on the importance score and semantic similarity. This proposed re-ranking algorithm improves the search engine result.

References

- [1] Wang, S. Jiang, Y. Zhang, M. Wang, "Re-ranking search results using semantic similarity", in Proc.FSKD, 2011, pp.1047-1051.
- [2] Zhi Lu, Tianyong Hao, Liu Wenyin, "Categorizing and ranking search engine's results by semantic similarity", international conference on Ubiquitous information management and communication 2009.
- [3] Abdelkrim Bouramoul, Mohamed-Khireddine Kholadi, Bich-Liên Doa, "An ontology-based approach for semantic ranking of the web search engines results".
- [4] Andre Freitas, Edward Curry, Sean O'Riain, "A Distributional Approach for Terminological Semantic Search on the Linked Data Web", SAC'12 March 25-29, 2012
- [5] Hao, T.Y., Hu, D.W., Liu, W.Y., Zeng, Q.T., "Semantic patterns for user-interactive question answering", Concurrency and Computation: Practice and Experience, Vol. 20, pp. 1-17, 2007.
- [6] George Tsatsaronis, Iraklis Varlamis, Kjetil Nørvag "SemanticRank: Ranking Keywords and Sentences Using Semantic Graphs", COLING'2010, Beijing, China, August 2010.
- [7] C. Leacock, M. Chodorow, "Combining local context and wordnet similarity for word sense identification", In WordNet: An Electronic Lexical Database, pp. 265-283, 1998.
- [8] [Online] Available: <http://www.wordnet.princeton.edu/>



Rutuja Ajmire received her B.E. degree in Computer Science & Engineering from HVPM College of Engineering and Technology, Amravati, Maharashtra, India, in 2007, Pursuing M.Tech. degree in Computer Science and Engineering from Government college of Engineering, Amravati, Maharashtra, India. Her research interests include Search Engine, Semantic Web.



India.

A. V. Deorankar has received his M.E. degree in Electronics Engineering from Govt. College of Engineering, Amravati, India. He has published nine papers in national level and seven papers in international journals. He has also patent of one paper. His area of research includes Computer Network, Web Mining. Currently he is working as an Associate Professor at Govt. college of Engineering, Amravati, Maharashtra,



Maharashtra, India.

P. N. Chatur has received his M.E. degree in Electronics Engineering from Govt. College of Engineering, Amravati, India and PhD degree from Amravati University. He has published twenty papers in national and ten papers in international journal. His area of research includes Neural Network, data mining. Currently he is head of Computer Science and Engineering department at Govt. College of Engineering, Amravati,