TOWARDS A SEMANTIC SEARCH MECHANISM BASED ON QUERY EXPANSION

Manh Hung Nguyen[1,*], Tan Hiep Nguyen[2]

[1]Post and Telecommunication Institute of Technology (PTIT), Hanoi, Vietnam.

[2]Vietsoftware International Corporation, Hanoi, Vietnam

**Email: nmhufng@yahoo.com,*

Received: 31 January 2013. Accepted for publication: …….. 2013.

ABSTRACT

Most of current search engines are syntax matching based. This may, in some circumstances, lead the search results having some irrelevant records regarding the input keywords. In order to limit this case from occurring, this paper presents a model for semantic search engine. This engine is based on normal search engines by adding an immediate level of semantic processing which makes the processing become transparent to users through three phases. The first phase is to generate a set of semantic relevant keywords from the original keywords based on their semantic relations defined in WordNet. The second phase is to give these semantic relevant keywords to some normal search engines such as Google, Yahoo, etc... search engine and then automatically collect their results based on hidden web techniques. The final phase is to re-rank the results from these normal search engines regarding the relation from each of them to the related semantic keywords, and the relation from the semantic keywords to the original keywords. This phase could be done by using some semantic similarity measurements.

The experiment results from this semantic search engine indicate some significant differences from normal search engines' results.

Keywords: Semantic search engine, Semantic similarity, Semantic ranking, Search engine, Hidden web, WordNet.

I. INTRODUCTION

The World Wide Web (WWW) is now growing up too fast. There are a huge data deposited on the WWW each day. This increases the difficulties when searching needed information in such a jungle of data. In such a context, the classical syntax search engines are no more suitable because they bring the search results which are too large and less relevant to the requirement. The growth of a new generation of search engines, called *semantic search engine*, is considered as an evident tendency.

At the level of modelling, there are several semantic search engines proposed. For instances, Lamberti et al. [1] proposed a relation-based page rank algorithm to be used in conjunction with semantic Web search engines that simply relies on information that could be extracted from user queries and on annotated resources. Relevance is measured as the probability that a retrieved resource actually contains those relations whose existence was assumed by the user at the time of query definition. Delbru et al. [2] proposed to exploit locality on the Web of Data by taking a layered approach, similar to hierarchical PageRank approaches. They provide justifications for a two-layer model of the Web of Data, and introduce DING (Dataset Ranking) a novel ranking methodology based on this two-layer model. DING uses links between datasets to compute dataset ranks and combines the resulting values with semantic-dependent entity ranking strategies. Ning et al. [3] presented RSS - a framework for enabling ranked semantic search on the semantic web. In this framework, the heterogeneity of relationships is fully exploited to determine the global importance of resources. In addition, the search results can be greatly expanded with entities most semantically related to the query, thus able to provide users with properly ordered semantic search results by combining global ranking values and the relevance between the resources and the query. Hao et al. [4] obtained nouns and verbs from snippets obtained from search engine using Name Entity Recognition and part-of speech. A semantic similarity algorithm based on WordNet is proposed to calculate the similarity of each snippet to each of the pre-defined categories. A balanced similarity ranking method combined with Google's rank and timeliness of the pages is proposed to rank these snippets.

At the level of application, there are also many semantic search engine applications. For instances, the semantic search engines of Google, Hakia, Bing, Lexxe, Duckduckgo, Yahoo! Search, etc.

This paper presents a model for semantic search engine. This engine is based on normal search engines by adding an immediate level of semantic processing which makes the processing become transparent to users through three phases. The first phase is to generate a set of semantic relevant keywords from the original keywords. The second phase is to give these semantic relevant keywords to some normal search engines and then automatically collect their results. The final phase is to re-rank the results from these normal search engines regarding the relation from each of them to the related semantic keywords, and the relation from the semantic keywords to the original keywords.

The paper is organized as follow: Section II presents the semantic search engine; Section III evaluates the proposed engine regarding current search engines; and the conclusion is in Section IV.

II. SEMANTIC SEARCH ENGINE

This section presents our proposal semantic search engine: Section 2.1 presents general architecture of the semantic search engine; Section 2.2 presents the first phase which generates the related keywords from original entered keywords; Section 2.3 presents the second phase which gives these related keywords to normal search engines and then automatically collects their results; Section 2.4 presents the third phase which re-ranks the results from these normal search engines to obtain the final ranked results.

2.1 General architecture

The general architecture of the semantic search engine based on normal search engines is presented in Fig.1. There are three main phases. The first phase is to generate a set of semantic relevant keywords from the original keywords. This phase takes user typed keywords as input data, then, using some mechanisms to find all related words of the given original keywords set. At the output, there are a set of *semantic related keywords sets* which has a semantic relation with the original keywords set. These output sets are then passed to the second phase. And for each *semantic related keywords set*, there is a *semantic distance* to the original keyword set. This semantic distance is estimated based on some mechanisms to estimate the distance between two objects or words (Lin [5], Paolucci et al. [6], Ludwig and Reyhani [7], Tran and Nguyen [8], Wang et al. [9]).

The second phase is to give these semantic relevant keywords to some normal search engines and then automatically collect their results. This phase is automatically done by using some techniques of hidden web (Raghavan and GarciaMolina [10], Madhavan et al. [11], Liddle et al. [12]). These techniques enable to automatically fill a form on any web page and then, retrieve the results without need the operation of user.

The final phase is to re-rank the results from these normal search engines regarding the relation from each of them to the related semantic keywords, and the relation from the semantic keywords to the original keywords. This phase could be done by using some semantic similarity measurements. The next sections will present these phases in detail.

2.2 Phase 1: Generation of related keywords

Let $O = \{o_1, o_2 ... o_n\}$ be a set of original keywords. For each original keyword $o_i, i = 1..n$, there is a set of semantic related keywords $S^i = \left\{s_1^i, s_2^i, ... s_m^i\right\}$. A set $R = \{r_1, r_2 ... r_n\}$ is called a *semantic related set* of O if and only if $r_1 \in S^1, r_2 \in S^2, .. r_n \in S^n$.

The *semantic distance* between the original set $O = \{o_1, o_2 ... o_n\}$ and the semantic related set $R = \{r_1, r_2 ... r_n\}$ is calculated based on the semantic distance between each pair of two keywords in two sets [8]:

$$D(O,R) = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} d(o_i, r_j) \qquad (1)$$

where $d(o_i, r_j)$ is the semantic distance between two keywords o_i and r_j. This distance could be detected based on an ontology as in the work of Tran and Nguyen [8] or based on WordNet (Fellbaum [13]) as follow:

- Without loss of generality, for each original keyword o_i,i = 1..n, there is a set of (ordered) semantic related keywords $S^i = \left\{s^i_1, s^i_2, \ldots s^i_m\right\}$ generated based on WordNet: if $k < j$ then s^i_k is semantically closer o_i than s^i_j.
- The semantic distance from s^i_j to o_i is: $d(s^i_j, o_i) = \frac{j-1}{|S_i|}$
- The semantic distance $D(R_l, O)$ from any word $w \notin S^i$ to o_i is 1: $d(w, o_i) = 1$. At the end of phase 1, we have a set of *semantic related set* $\{R_1, R_2 \ldots R_k\}$, $k = \prod_{i=1}^{n} |S^i|$, and each related set R_l has a semantic distance to the original keywords set. These data will be reserved for re-ranking procedure in the second and the third phase.

2.3 Phase 2: Automatic filling & getting results

From $k = \prod_{i=1}^{n} |S^i|$ sets of *semantic related set*$\{R_1, R_2 \ldots R_k\}$ which are generated from the original keywords set O. We fill each *semantic related set* R_l as input of normal search engines such as Google's search, Yahoo's search, etc. This step is automatically done by using hidden web techniques. This technique enables to automatically fill some keywords into a form of any web site, and then automatically retrieve the results as those displayed when user types the corresponding keywords (Raghavan and Garcia-Molina [10], Madhavan et al. [11], Liddle et al. [12]).

The obtained results, at the output of normal search engine, of the *semantic related set* R_l is an ordered set of pages $P^l = \{p^l_1, p^l_2 \ldots p^l_z\}$: if $i < j$ then p^l_i is ranked (by normal search engines) before p^l_j. The results for all *semantic related sets* from this phase will be considered to re-rank in the third phase.

2.4 Phase 3: Re-ranking

In order to obtain the final rank of all pages found from the phase 2, we calculate the *score* of each page regarding two aspects:

- The *rank* of this page in the results of the corresponding *semantic related set* R_l. The more this rank is high, the more the final rank of this page is high.
- The *semantic distance* from the *semantic related set* R_l to the original keywords set O. The more this distance is low (the two sets are closer), the more the final rank of this page is high.

Assume that the obtained results of the *semantic related set* R_l is an ordered set of pages $P^l = \{p^l_1, p^l_2 \ldots p^l_z\}$: if $i < j$ then p^l_i is ranked (by normal search engines) before p^l_j. First, the *score by rank* of page p^l_i, denoted sr^l_i, is calculated by choosing a limit number *MAX* of result pages which could be differentiated their *score by rank* (In the experiment, we set *MAX* = 100):

$$sr^l_i \begin{cases} \frac{MAX-i}{MAX} & if\ 1 \le i \le MAX \\ 0\ if\ i \ge MAX \end{cases} \quad (2)$$

Second, the *score by semantic distance* of page p^l_i, denoted ss^l_i, is calculated as follow:

$$ss^l_i = 1 - D(R_l,O) \quad (3)$$

where $D(R_l,O)$ is the semantic distance from the related keywords set R_l to the original keywords set O.

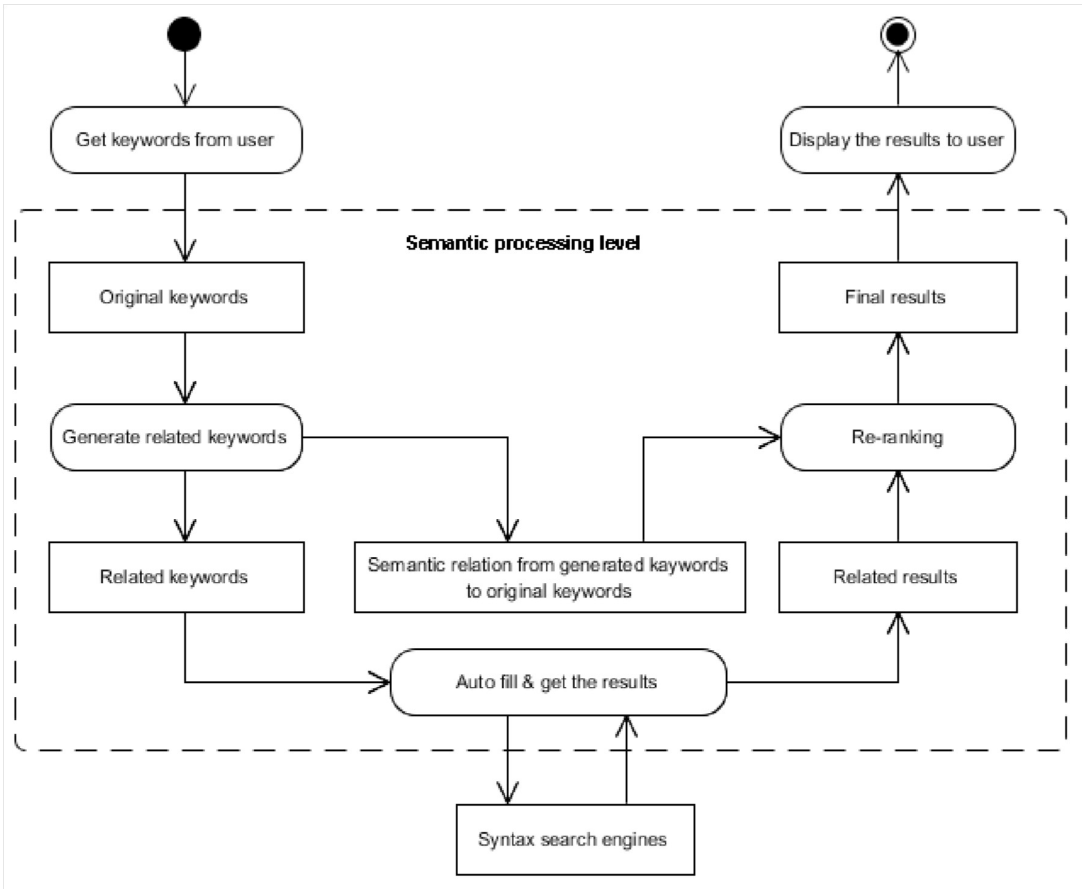
Get keywords from user
Display the results to user
Semantic processing level
Original keywords
Final results
Generate related keywords
Re-ranking
Related keywords
Semantic relation from generated kaywords to original keywords
Related results
Auto fill & get the results
Syntax search engines

Figure 1. The semantic search engine is considered as an extension of syntax search engines by adding a semantic processing level

The *final score* of page p^l_i, denoted $score^l_i$, is defined as follow:

$$score^l_i = f(sr^l_i, ss^l_i) \quad (4)$$

where f is a mapping: $[0,1]^2 \to [0,1]$ which satisfies these conditions:

1. $f(1,1) = 1$
2. $f(0,0) = 0$
3. $f(r_1,s) \leq f(r_2,s)$ if $r_1 \leq r_2$

4. 4. $f(r,s_1) \le f(r,s_2)$ if $s_1 \le s_2$

For instance, in our experiment, we use the function $f(r,s) = r * s$.

The *final rank* of page p^l_i is the position of $score^l_i$ ranked for $\forall i,\forall l$, from the biggest to the smallest.

Summarily, the steps-by-steps of processing in our proposed semantic search engine are depicted in Algorithm.1: the input keywords are captured from the user (line 1). Then, each words in this original keywords set is considered to generate all semantic related words (lines 2-4).

Algorithm 1 Step-by-step in the proposed semantic search engine

Input: a keywords set entered by user

Output: a set of result pages ranked from the highest score to the lowest score

```
 1: $O \leftarrow$ the keywords set entered by user
 2: for all keyword $o_i \in O$ do
 3:     $S^i \leftarrow generate(o_i)$
 4: end for
 5: $R \leftarrow \varnothing$
 6: $index \leftarrow 1$
 7: for all related keywords set $S^i$ of $o_i \in O$ do
 8:     $R_{index} \leftarrow$ union of an element $s^i_j \in S^i, \forall i : 1..n$
 9:     $index \leftarrow index + 1$
10:     $R \leftarrow R \cup R_{index}$
11: end for
12: for all $i$ from 1 to $index$ do
13:     $P^i \leftarrow normalSearch(R^i)$
14:     $ss^i \leftarrow scoreBySemantic(O,R^i)$
15:     for all page $p^i_j \in P^i$ do
16:         $sr^i_j \leftarrow scoreByRank(p^i_j,P^i)$
17:         $score^i_j \leftarrow f(sr^i_j,ss^i)$
18:     end for
19: end for
20: $P \leftarrow \cup P^i, \forall i : 1..index$, ranked by $score^i_j$ of each $p^i_j \in Pi$
21: return $P$
```

From these related keywords sets of each original keyword, a set of *semantic related keywords sets* are combined (lines 5-11). This is the end of phase 1. Phase 2 takes its turn by entering each *semantic related keywords set* into a normal search engine and then, getting the output results in format of pages (line 13). At the same time, phase 3 starts by calculating the *score by semantic* of each *semantic related keywords sets* regarding its original set (line 14). And then it calculates the *score by rank* for each page in the results set (line 16). After that, the *final score* of each result page is calculated (line 17). The final results are obtained after ranking all results pages by their final score, from the highest to the lowest (lines 20-21).

III. EVALUATION

In this section, we take an experiment to compare the proposed model with some normal search engines. Section 3.1 presents the setting up of experiment; section 3.2 presents and analysis the obtained results.

3.1 *Experiment setup*

We implement the proposed search engine based on the search engine of Google's search, and using parameters as indicated in Table 1.

The scenario of the experiment is:

- Step 1. Initiation of original keywords sets: we use 1000 original sets of keywords, each set has a different size from 1 keywords to 5 keywords. These original keywords sets are stable for all tests.

Table 1. Experement setup parameters

Parameters	Values
Number of original keywords sets	1000
MAX (limit number of results considered)	100
Normal search engine	Google's search
Mapping *f*(*r*,*s*)	$r * s$
Number of pages to compare	10 | 20 |50 | 100

- Step 2. Running with normal search engine: For each original keywords set, enter it into Google's search page, and then collect the results in the format of pages: 10, 20, 50, and 100 first pages.
- Step 3. Running with proposed search engine: With the same original keywords set, enter it into proposed search engine, and then collect the results in the format of pages: 10, 20, 50, and 100 first pages.
- Step 4. Comparing the obtained results from two search engines regarding 10, 20, 50, and 100 first pages by calculated the number of new pages obtained from step 3 regarding the ones obtained from step 2.
- Repeat from step 2 to step 4 until all original keywords sets are passed both search engines and their results are saved.
- Step 5. Analysing the statistics from step 4 after 1000 times of comparison on each 10, 20, 50, and 100 first pages.

3.2 ***Results***

The results are presented in Fig.2. There are significant differences between the results of semantic search engine and those of normal search engine. For instances, considering of 10 first obtained pages, there are about 20% of pages are new from the results of semantic search engine. In the same line with it, there are about 29%, 46%, 58% of pages are new from the results of semantic search engine, if we consider 20, 50, and 100 first obtained pages, respectively.

The results indicate that in general, using our *semantic processing level* on the search engine of Google could be better than the original search engine. The bigger the number of results considered, the better our model regarding the original search engine.

However, there is also a limit of this approach. That is the limitation in automatic filling the form of existing search engines. This may prevent this model from applying into reality.

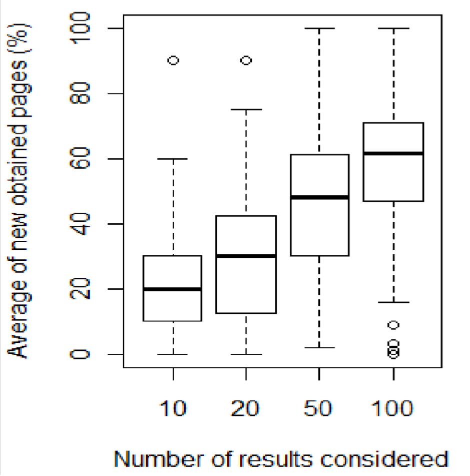

Figure 2. Percentage of number of new pages in the results of semantic search engine compared to those of normal search engine, regarding on different numbers of considered pages

IV. CONCLUSION

This paper presented a model for semantic search engine. This engine is based on normal search engines by adding an immediate level of semantic processing which makes the processing become transparent to users through three phases. The first phase is to generate a set of semantic relevant keywords from the original keywords. The second phase is to give these semantic relevant keywords to some normal search engines and then automatically collect their results. The final phase is to re-rank the results from these normal search engines regarding the relation from each of them to the related semantic keywords, and the relation from the semantic keywords to the original keywords. The experiment results from this semantic search engine indicate some significant differences from normal search engines' results.

Improvement of techniques to estimate the semantic distance in phase 1, using different normal search engines in phase 2, and comparing the results with different semantic search engines are some of our research works in the near future.

REFERENCES

[1] Fabrizio Lamberti, Andrea Sanna, and Claudio Demartini. A relation-based page rank algorithm for semantic web search engines. *IEEE Trans. on Knowl. and Data Eng.*, 21(1):123 – 136, January 2009.

[2] Renaud Delbru, Nickolai Toupikov, Michele Catasta, Giovanni Tummarello, and Stefan Decker. Hierarchical link analysis for ranking web data. In *Proceedings of the 7 th international conference on The Semantic Web: research and Applications - Volume Part II*, ESWC'10, pages 225–239 , Berlin, Heidelberg, 2010. Springer-Verlag.

[3] Xiaomin Ning, Hai Jin, and Hao Wu. RSS: A framework enabling ranked search on the semantic web. *Inf. Process. Manage.*, 44(2):893–909, 2008.

[4] Tianyong Hao, Zhi Lu, Shitong Wang, Tiansong Zou, Shenhua GU, and Liu Wenyin. Categorizing and ranking search engine's results by semantic similarity. In *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, ICUIMC '08, pages 284 – 288, New York, NY, USA, 2008. ACM.

[5] Dekang Lin. An information-theoretic definition of similarity. In *Proc. 15th International Conf. on Machine Learning*, pages 296–304. Morgan Kaufmann, San Francisco, CA, 1998.

[6] Massimo Paolucci, Takahiro Kawamura, Terry R. Payne, and Katia P. Sycara. Semantic matching of web services capabilities. In *Proceedings of the First International Semantic Web Conference on The Semantic Web*, ISWC '02, pages 333–347 , London, UK, 2002. Springer-Verlag.

[7] Simone A. Ludwig and S. M. S. Reyhani. Semantic approach to service discovery in a grid environment. *Journal of Web Semantic*, 4(1):1–13, 2006.

[8] Dinh Que Tran and Manh Hung Nguyen. A mathematical model for semantic similarity measures. *South-East Asian Journal of Sciences*, 1(1):32–45, 2012.

[9] Gongzhen Wang, Donghong Xu, Yong Qi, and Di Hou. A semantic match algorithm for web services based on improved semantic distance. In *Proceedings of the 2008 4 th International Conference on Next Generation Web Services Practices*, pages 101–106, Washington, DC, USA, 2008. IEEE Computer Society.

[10] Sriram Raghavan and Hector Garcia-Molina. Crawling the hidden web. In *Proceedings of the 27th International Conference on Very Large Data Bases*, VLDB '01, pages 129 – 138, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[11] Jayant Madhavan, David Ko, Lucja Kot, Vignesh Ganapathy, Alex Rasmussen, and Alon Halevy. Google's deep web crawl. *Proc. VLDB Endow.*, 1(2):1241–1252, August 2008.

[12] Stephen W. Liddle, Sai Ho Yau, and David W. Embley. On the automatic extraction of data from the hidden web. In *Revised Papers from the HUMACS, DASWIS, ECOMO, and DAMA on ER 2001 Workshops*, pages 212–226, London, UK, UK, 2002. Springer-Verlag.

[13] C. Fellbaum. *WordNet: An Electronical Lexical Database*. The MIT Press, Cambridge, MA, 1998.

TÓM TẮT

HƯỚNG ĐẾN MỘT CƠ CHẾ TÌM KIẾM NGỮ NGHĨA DỰA TRÊN MỞ RỘNG TRUY VẤN

Nguyễn Mạnh Hùng
Học viện công nghệ bưu chính viễn thông (PTIT)
Hà Nội, Việt Nam
Email: nmhufng@yahoo.com
Nguyễn Tấn Hiệp
Công ty Vietsoftware International
Hà Nội, Việt Nam
Email: tahi1990@gmail.com

Hầu hết các công cụ tìm kiếm hiện tại được dựa trên cú pháp phù hợp. Trong một số trường hợp, điều này có thể dẫn đến những kết quả tìm kiếm có một số bản ghi không liên quan với từ khóa đầu vào. Để hạn chế trường hợp này xảy ra, bài báo này trình bày một mô hình công cụ tìm kiếm ngữ nghĩa. Công cụ này được dựa trên các công cụ tìm kiếm thông thường bằng cách thêm một tầng trên trực tiếp xử lý ngữ nghĩa làm cho việc xử lý trở nên trong suốt đối với người sử dụng qua ba giai đoạn. Giai đoạn đầu tiên là tạo ra một tập ngữ nghĩa các từ khóa có liên quan từ những từ khóa ban đầu dựa trên các quan hệ ngữ nghĩa của chúng được định nghĩa trong WordNet. Giai đoạn thứ hai là đưa các từ khóa có liên quan ngữ nghĩa vào một số công cụ tìm kiếm thông thường như Google, Yahoo, vv .. và sau đó tự động thu thập các kết quả dựa trên các kỹ thuật web ẩn. Giai đoạn cuối cùng là xếp hạng lại các kết quả từ các công cụ tìm kiếm thông thường dựa trên mối quan hệ từ mỗi từ trong số đó với từ khóa ngữ nghĩa liên quan, và mối quan hệ từ những từ khóa ngữ nghĩa với các từ khóa ban đầu. Giai đoạn này có thể được thực hiện bằng cách sử dụng một số phép đo độ tương tự.

Các kết quả thí nghiệm từ công cụ tìm kiếm ngữ nghĩa cho thấy một số khác biệt đáng kể từ kết quả của các công cụ tìm kiếm thông thường.

Từ khóa. công cụ tìm kiếm ngữ nghĩa, độ đo tương tự, xếp hạng ngữ nghĩa, công cụ tìm kiếm, web ẩn, WordNet.