Volume 3, Issue 4, April 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Lexical Ambiguity Resolution in Web Search for Phrases

Rekha Jain[1] **Sulochana Nathawat**[2] **Rupal Bhargava**[3] **G.N. Purohit**[4]

Department of Computer Science, Banasthali University[1, 2, 3, 4]

India

Abstract*— Web Search Engine requires the effectiveness and efficiency of response time and results. Searching on Web returns thousands of results. The task of Search Engine is to find relevant information according to user's information need. Some irrelevant results occur due to presence of ambiguous keywords. To remove this ambiguity we are presenting a system that considers the user's need, modifies the queries and gives the most relevant results. Our Proposed system provides a layer on to the Google Search Engine for retrieving the most relevant results at top position.

Keywords*— Web Search Engine, Lexical ambiguity, Word Sense Disambiguation, Page Rank Algorithm, Information Retrieval.

I. Introduction

The World Wide Web is the huge collection of heterogeneous documents. Web Search Engine is designed to search information on the World Wide Web. There are various types of Search Engines available on the Internet. Information may consist of images, audio, video, web pages, text documents etc. A word, phrase or sentence is ambiguous if it has more than one interpretation. In this paper we concern about the ambiguous phrases. Ambiguous phrase generates the two or more incompatible and unrelated meanings. Lexical ambiguity of word or phrase can be used to express two or more different meanings. Disambiguation is done on the basis of syntax, word meaning, context and knowledge of the world. Word Sense Disambiguation (WSD) is defined as the task of identifying the sense of word in textual context.

The structure of this paper is as follows: section 2 provides the brief overview of Web Search Engine, section 3 describes the Lexical Ambiguity, section 4 describes the Word Sense Disambiguation and its approaches, section 5 shows the proposed Dynamic Page Rank Algorithm, section 6 discusses the results, and section 7 summarizes the Conclusion.

II. Web Search Engine

Web Search Engines crawl the Web, download and index the pages in order to allow full text searching. Fig. 1 shows the architecture of typical Search Engine. User's information need is expressed in term of keywords or phrases. Query is passed to the database and results are ordered and shown to the user. Single ordering policy is used by Search Engine i.e. if same query is passed by many users they all get the same results presented in the same order [1]. Search Engines give the results on the bases of keywords not according to the textual context.

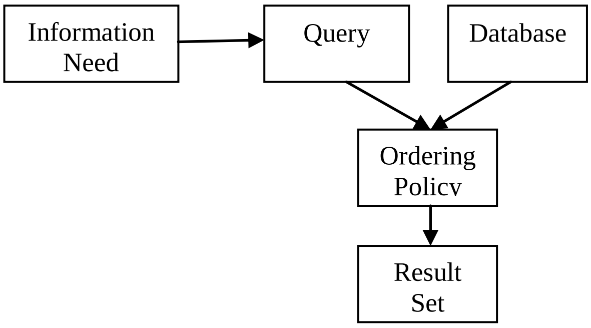

Fig. 1 The architecture of standard Search Engine

Search Engine contains three main components crawler, indexer, Ranker [2].

A. Crawler

Crawler is also known as spider. Web crawler is a program that downloads and stores Web pages. Collected pages are used by Web search engines for indexing. Crawler's task is to find the information on the hundred of millions of Web pages. Crawler builds the list of words found on websites. The process of building the list of words is called the Web Crawling.

B. Indexer

Everything the spider finds goes into the second part of Search Engine i.e. index. Indexer is a program that reads the pages downloaded by spider. Search engine index contains the data about URL when searching is performed. Indexed information is saved in database. The purpose of indexing the documents is the fast and accurate retrieval of relevant results for search query.

C. Ranker

When user enters the query it is interpreted by the search engine and find the relevant document recorded in the index. Ranker ranks these relevant document and presents to the user. Different ranking mechanisms are used for ranking the web pages.

III. Lexical Ambiguity

Ambiguity of word, phrase, or sentence is the ability to express more than one meaning. There are two kinds of ambiguity: lexical ambiguity and syntactic or structural ambiguity. In this paper we are dealing with lexical ambiguity of phrases. Lexical ambiguity is based on the single word having two or more possible meanings. Homophony and polysemy are two types of lexical ambiguity. Homophony occurs when a single word has more than one meaning. Homophony also occurs when a word not spelt same but pronounced same and has different meanings. For example the ambiguous phrase "Investment in the Banks". Here bank has more than one meaning in the sense of river or financial institution. Polysemy occurs when a single word has multiple related meaning. Meaning of polysemy word or phrase can be identified by the context in which it is used. For example the phrase "Types of Command" has more than one meaning in the sense of computer command and military command [3].

In syntactic ambiguity Sentence have more than one interpretation even if there is no ambiguous word. Reason behind this is the structure of the sentence i.e. its syntax. For example: I am prepared to give the sum of one million dollars to you and your husband. This can be understood as I am prepared to give the sum of (one million dollars) (to you) and (your husband) – making a total of two million dollars; or as I am prepared to give the sum of (one million dollars) to (you and your husband) - making a total of only one million dollars [4].

IV. Word Sense Disambiguation

Lexical ambiguity can be resolved by the methods that automatically assign the appropriate meaning of word in context. This task referred as Word Sense Disambiguation. WSD is the process of identifying the sense of word in textual context, when word has multiple meanings. WSD associate a word in a text or sentence having different meaning. There are two main approaches of WSD, Deep Approaches and Shallow Approaches. Deep approaches are based on world knowledge but shallow approaches do not use the world knowledge. Neighbouring words are used to identify the sense of words [5]. There are four methods of WSD [6] [7]:

A. Dictionary-based Approaches

The first implementation of this approach is done by Lesk [1986]. All the sense of the word to be disambiguated retrieved from the dictionary. Each sense is then compared to the dictionary definition of remaining word in context. The sense which meets the context word is chosen as sense.

B. Supervised Approaches

Supervised WSD uses machine learning techniques in which inputs are manually sense-annoted data and output is a classifier system. In this approach, a sense disambiguation system is learned from a representative set of labelled instances drawn from the same distribution as the test to be used. Generally supervised approaches give better results than unsupervised approaches.

C. Semi-supervised Approaches

The drawback of supervised approaches is the requirement of a large sense tagged training set. Bootstrapping approach does not require the large training data set. It works on few numbers of instances of each sense of target word. Initial classifier is trained using a labelled instance which is known as seed. The task of initial classifier is to extract large training set from the remaining untagged corpus. On repeating this process we will get a series of improved classifier.

D. Unsupervised Approaches

This approach uses the sense tagged data of any type during the training. The concept behind this technique is that the words which have same sense will also have similar neighbouring words. Input to this approach is unlabeled instances which are represented as feature vector. Then these are grouped into clusters based on similarity metric. From input text word sense can be assigned from cluster to which they are closest based on similarity metric.

V. Proposed Algorithm

Our proposed algorithm (Dynamic Page Rank algorithm) acts as a layer on to search engine. This algorithm resolves the ambiguities of polysemy words. Unlike Page Rank algorithm in which results are retrieved on the basis of ranks of web pages, Dynamic algorithm gives the more relevant results on the basis of dynamic page rank. Dynamic page rank is calculated using Google's page rank. Our algorithm works for all Search Engines.

Input: Search Terms

Output: Relevant Results

Proposed Algorithm

```
Step 1: Read the search terms.
Step 2: Apply Tokenization, stemming and removal of stop words.
Step 3: If the token is ambiguous then apply the correct sense.
Step 4: Pass the modified query to the database.
```

Step 5: Results retrieved on the basis of page rank assigned by Google's Page Rank Algorithm.
Step 6: Generation of the dynamic page ranks of Web pages that are the part of result set.
Step 7: Rearrange the results on the basis of dynamic page rank.

Fig. 2 shows the flowchart of proposed Dynamic Page Rank Algorithm.

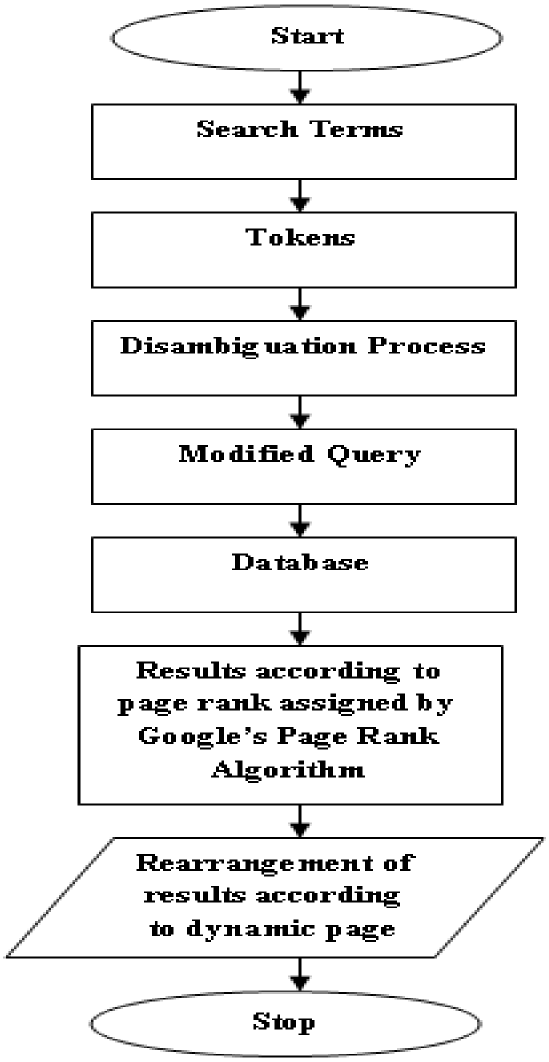
Start
Search Terms
Tokens
Disambiguation Process
Modified Query
Database
Results according to page rank assigned by Google's Page Rank Algorithm
Rearrangement of results according to dynamic page
Stop

Fig. 2 Flowchart of Proposed Algorithm

VI. EXPERIMENTAL RESULTS

We have applied Reciprocal Rank (RR) and Average Precision (AP) to compare the efficiency of Page Rank algorithm and Dynamic Page Rank algorithm. Reciprocal rank is the inverse of the rank of first correct answer [8].

$$Reciprocal\ Rank = \frac{1}{rank} \qquad (1)$$

Precision is the fraction of retrieved documents that are relevant and recall is the fraction of relevant instances that are retrieved. A measure that uses both precision and recall is the average precision. Precision as a function of recall is denoted by p(r). Average Precision computes the average value of p(r) over the interval r=0 to r=1 [9].

$$AveP = \sum_{k=1}^{n} p(k)\,\Delta r(k) \qquad (3)$$

Where, k is the rank in the sequence of retrieved documents, n is the number of retrieved documents. Suppose user searches for ambiguous phrases "Types of Command" and "Length of String". These ambiguous phrases can be used in different senses. Like phrase "Types of Command" can be referred in the sense of type of computer commands or in the sense of types of military commands. Same phrase "Length of String" can be referred in the sense of type of data string used in computer languages or in the sense of string music. Our proposed system calculates the Reciprocal Rank and Average Precision using equations (1) and (2) respectively. Table 1 shows the reciprocal rank of Page Rank algorithm and Dynamic Page Rank algorithm. Table 2 shows the average precision of Page Rank algorithm and Dynamic Page Rank algorithm.

TABLE I
RECIPROCAL RANK OF PAGE RANK ALGORITHM AND PROPOSED ALGORITHM

Phrases	**Reciprocal Rank (Page Rank Algorithm)**	**Reciprocal Rank (Dynamic Page Rank Algorithm)**
Types of Command (Computer)	0.046	1
Types of Command (Military)	0.25	1
Length of String (Computer)	0.005	1
Length of String (Music)	0.167	1

TABLE II
AVERAGE PRECISION OF PAGE RANK ALGORITHM AND PROPOSED ALGORITHM

Phrases	**Average Precision (Page Rank Algorithm)**	**Average Precision (Dynamic Page Rank Algorithm)**
Types of Command (Computer)	0.101	0.289
Types of Command (Military)	0.155	0.395
Length of String (Computer)	0.005	0.007
Length of String (Music)	0.1	0.34

Fig. 3 shows the comparative results of Reciprocal Rank and Average Precision of Google’s Page Rank algorithm and Dynamic Page Rank algorithm and proves that Dynamic Page Rank algorithm provides much more relevant results than Page Rank algorithm.

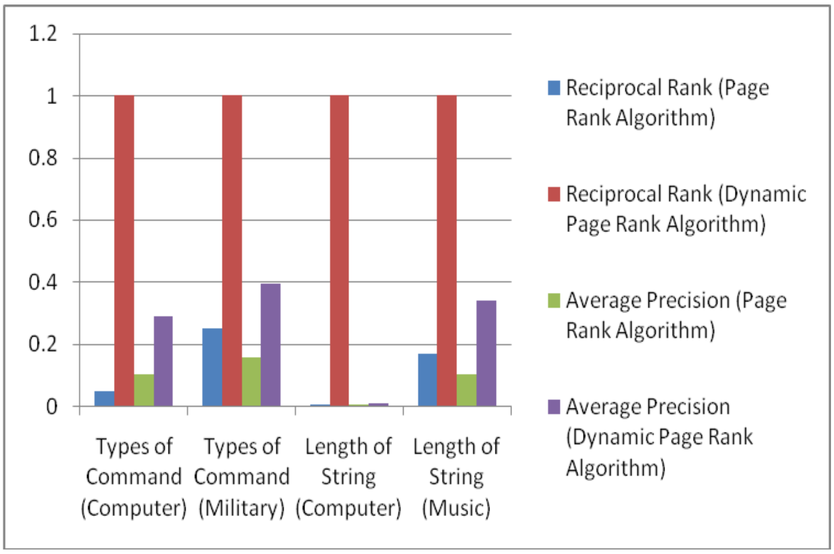

Fig. 3 Comparative Results of RR and AP

VII. CONCLUSIONS

Page Rank algorithm is widely used and most accepted algorithm. Results show that performance of Dynamic Page Rank algorithm is better than Google’s Page Rank algorithm. Dynamic Page Rank algorithm is able to solve the lexical ambiguity of ambiguous phrases and provides much more relevant content on the top of Search Engine. Page Rank algorithm never resolves the ambiguities. It already computes the page ranks of pages and store them in database. At the time of searching it simply arranges the pages from higher to lower rank values without considering their ambiguous behavior.

REFERENCES

[1] Eric J. Glover, Steve Lawrence, Michael D. Gordon, William P. Birmingham, C. Lee Giles, “Web Search – Your Way”.

[2] Monica Peshave, “How Search Engines Work and a Web crawler application”.

[3] Lexical Ambiguity. [Online]. Available: www.angelfire.com/tn/semantics/amblex.html

[4] The Ambiguity | DiploFoundation. [Online]. Available: www.diplomacy.edu › Language

[5] Esha Palta, “Word Sense Disambiguation”, M. Tech. dissertation, dept. CSE Indian Institute of Technology, Mumbai, 2006.

[6] Daniel Jurafsky, James H. Martin, *Speech and Language Processing*, 2nd ed., 2009, ch. 17, pp. 657-690.

[7] Rekha Jain, Sulochana Nathawat, “Sense Disambiguation Techniques: A Survey”, International Journal of Advances in Computer Science and Technology, Vol. 1, No. 1, pp. 1-6, 2012.

[8] Mean reciprocal rank. [Online]. Available: http://en.wikipedia.org/wiki/Mean_reciprocal_rank.

[9] Information retrieval. [Online]. Available: http://en.wikipedia.org/wiki/Information_retrieval.