Performance comparison of Word Sense Disambiguation (WSD) Algorithm on Hindi Language Supporting Search Engines

Parul Rastogi[1] and Dr. S.K. Dwivedi[2]

[1]Research Scholar, Computer Science Dept., BabaSaheb BhimRao Ambedkar University
Lucknow, UP, India

[2]Associate Professor, Computer Science Dept., BabaSaheb BhimRao Ambedkar University
Lucknow, India

Abstract

Search Engines are the basic tool of fetching the information on the web. The IT revolution not only affected the technocrats, but the native users are also affected. The native users also tend to look for any information on web nowadays. This leads to the need of effective search engines to fulfill native user's needs and provide them information in their native languages. The major population of India use Hindi as a first language. The Hindi language web information retrieval is not in a satisfactory condition. Besides the other technical setbacks, the Hindi language search engines face the problem of sense ambiguity. Our WSD method is based on Highest Sense Count (HSC). It works well with Google. The objective of the paper is comparative analysis of the WSD algorithm results on the three Hindi language search engines- Google, Raftaar and Guruji. We have taken a test sample of 100 queries to check the performance level of the WSD algorithm on various search engines. The results show promising improvement in performance of Google search engine whereas the least performance improvement was there in Guruji search engine.

Keywords: *Word Sense Disambiguation, Hindi language Search Engines, Sense Ambiguity, Precision*

1. Introduction

The unhampered growth of web as a complete reservoir of knowledge has lead in an era of Information Revolution. To date, the Internet is the foremost source of information for the human population. English is the most dominated and preferred language for the web access. In recent times, the rapid growth in the popularity of computers and the Internet in non-English speaking countries like India, have increasingly made the need and importance of reaching out to the non-English speaking zone. With the increase in contents written in native languages on the Internet, a proper mechanism is needed to make this content noticeable and available wherever and whenever necessary.

The major population of India use Hindi as a first language. Hindi-IT market seems to have taken-off silently during the past couple of years. Today, not only Hindi language, but also other Indian languages such as Tamil have begun to be noticed by IT bigwigs. The Internet penetration growth rate is among the fastest in the world in India. But everyone knows its going to hit a plateau if the main Internet language remains English. Sure enough, there are portals like Rediff, Yahoo, MSN, Google and others have started offering contents in Hindi and other languages. But, hunting for any real information, it seems like the amount of activity that a person can do on the Internet with Hindi or any other local language, is limited. The serious concern should be taken by the Indian IT industry to promote the web usage by the native speakers of India. This is possible only when the search engines would provide information to the native users in their languages only.

Various search engines are available on the Internet as independent search engine sites in English, but very few ([1]Google, [2]Raftaar, [3]Webkhoj, [4]Guruji etc.) Hindi language search engines are available. The performance of the existing search engines is not up to the mark. The search engines that support Hindi language search are not able to provide quality results. There are various problems, the search engines face with Hindi language information retrieval. Sense ambiguity is one of the major problems in information retrieval on the web in Hindi Language. Many words are polysemous in nature. Identifying the appropriate sense of the words in the given context is a difficult job for the search engines. WSD gives solution to the many natural language processing systems including information retrieval.

2. Related Work

Our work for disambiguation is focused on Hindi language web IR. The work is motivated by the [1]. They used the Total sense score (TSS) for the disambiguation. Our approach also finds the context in the similar way. In addition to their work for TSS we used the phrase frequency (PF) also. Phrase frequency is the occurrence of the complete query phrase in the retrieved resultset. By using it we are able to give better results. Besides that various other researchers have also used web documents for the disambiguation approach. [2].

The similar kind of work is performed by [3] to improve the web search results. Their work faces the problem of higher computational cost. The most time-consuming phase of their approach is the construction of the query graph, which requires intensive querying of the database of co-occurrences calculated from the Web1T corpus

The query expansion approach is based on Vector Space Model (VSM). The common specification about the VSM is term frequency and inverse document frequency. In our

[1] www.google.com
[2] www.raftaar.com
[3] www.webkhoj.iiit.ac.in
[4] www.guruji.com

study, we emphasize that query expansion is related to the terms in a relevant document itself. Using WordNet or the Web as whole for the query expansion is not a feasible solution for the query expansion in WSD. Since sense disambiguation is dependent on the context of the terms therefore it is quiet justified that the context of the query terms can be identified very well from the relevant document set only.

We have taken a base of Pintos approach used for the query expansion for WSD. Pinto's method's [4] success rate is low in improving the performance of IR system. One of the reason as mentioned earlier is the researcher used WordNet as the base for the query expansion which is a lexical database of contextual relations. Since web is a huge pool of information so in the case of web many times it is not feasible to find the context of the key terms with the existing examples of a lexical database. It is important to find out the current (in a particular query) context of a query from the set of relevant document set retrieved.

Jian-Yun and Jin [5] used the approach to query expansion which is different from most previous studies. They argue that an appropriate combination of the expansion terms with the original terms is an important problem to deal with in query expansion. Which has been taken as granted in the other research work that expansion terms should be added as additional dimensions in the resulting vector (e.g. in [6] and [7]). Nie and Jin's preliminary results seem to support the claim that considering the expansion terms as logical alternatives is a better solution. They also supported the fact that the WordNet is not a suitable resource for query expansion in information retrieval.

The similar kind of work which is also comparable to the tentative of [8] that tries to create more complex relationships within vectors. Wong et al. observed that the underlying independence assumption in vector representation is not reasonable. They suggest considering dependencies between dimensions in a Generalized Vector Space Model. However, the method of Wong et al. suffers from the complexity problem. In practice, it is difficult to fully implement it.

3. Sense Ambiguity and Hindi Language Web IR

Sense ambiguity in Hindi language queries can be clearly understood by the given example query "गुलाब की कलम {Gulaab ki kalam} (Rose branch)" (in Hindi language) consists of three terms as follows:

Terms Hindi	Senses from WordNet	POS (part of speech)
गुलाब{Gulaab} (Rose)	गुलाब{Gulaab}(Rose)	Noun
की {Ki}(of)		Preposition
कलम	पेन {Pen}(Pen), अँखिया {Ankhiyaan}, कलम {Kalam}, तूलिका {Tulika}(Brush), कलम {Kalam}(Branch), लेखनी {Lekhni}	Noun

It is unclear from the above mentioned query whether the user is interested in the कलम as a pen, कलम as a brush or कलम as a branch. Here कलम is a polysemous word. Before we resolve the ambiguity in query the first step should be the identification of the ambiguity level in the query. Our approach begins with the first step of ambiguity detection and finally to resolve query ambiguity, we implemented WSD algorithm.

4. Ambiguity Detection

The focus of the ambiguity detection method is to measure the ambiguity of a query term q*i* from a query Q. The low probability tagging is likely to be ambiguous. In the early phase of our research work we formulated an approach for the ambiguity detection based on two parameters "entropy" and "threshold" [9]. If the value of entropy is greater than threshold or we can say entropy passes a threshold, the query will be an ambiguous query. Detecting the ambiguity using the concept of entropy and threshold is found quite successful for Hindi language information retrieval. Ambiguity detection improves the performance of the WSD based applications. It reduces the overload on the system by avoiding the useless efforts to disambiguate the unambiguous queries (that is a query having polysemous words). The ambiguity resolution provides a robust mechanism for presenting results to a user for better conception of the contents of the result set.

5. Word Sense Disambiguation Approach

Subsequent to ambiguity detection next step is for word sense disambiguation. We have used the concept of Highest Sense Count (HSC) which is motivated from [1] approach. Their approach used the sum of frequencies of the meronym, synonym and holonym terms to evaluate the Total Sense Score (TSS). In our approach we used the (Sense Count) SC which simply uses the concept of counting the occurrence of the terms in a snippet in context of particular sense in a hypernyms, gloss sentences and test corpus. We had designed our own test corpus of example sentences in Hindi language. We calculated the Phrase Frequency which counts the occurrence of query phrase in snippets and also in hypernym, gloss and also in test corpus. We had calculated the HSC with the help of equation 1 which is as follows-

$$HSC = \max\left\{\forall_{i=0,n} PF_i / SC_i\right\} \quad \text{- Equation 1}$$

Here

Sense Count - $SC_i = \sum FT_{i=0}^{n}$

PF_i = Phrase frequency (is a count of query phrase in snippets and also in hypernym, gloss and test corpus).

FT_i = Frequency of T_i in Hypernym, Gloss and Test corpus.

T = Total Terms in snippets.

The highest value of HSC corresponding to a particular sense determines the context of that snippet. Following steps are used for the WSD method based on HSC-

Step -1 The top ten documents snippets are taken.
Step-2 Find the senses for a polysemous word.
Step-3 The frequency of each term T_i is calculated by counting its occurrence in the hypernym, gloss sentences and test corpus against each sense.
Step-4 The frequencies are summed up to find out the SCi for a particular sense.
Step-5 Similarly the phrase frequency PF_i is also calculated for the particular sense.
Step-6 Now using the equation -1 find the HSC for the snippet.
Step-7 The highest value of a HSC determines the context of a snippet in a particular sense.

The HSC helps in disambiguating the sense of a query and also facilitates to select the relevant document snippets from the top ten document snippets retrieved by the system.

6. Query Expansion based on Vector Space Model

6.1 Vector Space Model and Query Expansion

Prior to disambiguation next step is to retrieve more relevant resultset. This is possible only by regenerating query by query expansion. The Vector Space Model (VSM) has been a standard model of representing documents in information retrieval for almost three decades [10]. In VSM, all documents and queries are represented as vectors.

It is a fundamental model for web search engines. In VSM, the similarity between documents and the query is estimated by the inner product of the document vector and query vector. The similarity estimation is basically done by the distinctive terms in a document. Document vectors are weighted to enhance the effect of distinctive terms, that is, to give more importance to terms common in the document but rare in the collection as a whole. The more frequently and distinctively the query terms appear in the document, the more similar is the document with the query term. It is a pre-assumption in the VSM that query terms are self sufficient to compute the similarity between the document and the query. The search engine then presents to the user a list of links to the ranked documents which are contextually similar to the query. For single term queries this is often not the case. This means that the user may be presented with pages which are mostly only of marginal relevance. The user then has to scan many pages or iteratively refine or expand the query, adding terms to narrow down the retrieval result to the sort of pages user wants.

Query expansion is one of the techniques used to improve the performance of web information retrieval system. When query expansion technique is implemented on the web search engines, it involves evaluating the user's input and expanding the search query to match additional documents. Query expansion involves techniques such as: i) Finding synonyms of words, and searching for the synonyms as well; ii) Finding all the various morphological forms of words by stemming each word in the search query; iii) Fixing spelling errors and automatically searching for the corrected form or suggesting it in the results; and iv) Re-weighting the terms in the original query. Query expansion is a technology studied in the field of computer science, particularly within the realm of natural language processing and information retrieval.

Another technique can be used for the query expansion is to find the contextual key terms to expand the query especially in the case of WSD. The problem of sense ambiguity deteriorates the precision of the retrieved result set. Therefore, using the contextual terms for the query expansion is found one of the feasible solutions. We result into the increased number of relevant documents against the original query. It is a well-known fact that the terms which are related to the query exist in the same relevant document and also in the close proximity. The complete web document might is large enough. Therefore, we had taken snippets of the returned document set to identify the key terms that can be used to expand the original query.

6.2 Our approach of Query Expansion

Our query expansion approach is based on two parameters initial weight of the terms in a snippets and the distance of the highest weighted term with the query terms. We have utilized the snippets returned by the search engine as a source of document summary. The snippet helped clearly for the ambiguity detection approach and the relevant set of snippet (identified by WSD algorithm) is used further for the query expansion also. Our approach is quiet close to the [11] vector space relevance feedback method for the query expansion. Similar to Rocchio's approach, we also add terms for expansion from the relevant documents to the query.

So here the document vector D denotes the collection of snippets returned for the document (top ten documents) [12]. For the selected sense, the new document vector D' is created which is a subset of document vector D. This document vector D' will be used further for the query expansion.

In the next step, from the document vector D' we calculate the weight of each term in the document snippet. The initial weight is calculated by the simple formula of *Ctf***idf*, where *Ctf* is cumulative term frequency and *idf* is inverse document frequency.

Hence the formula for the initial weight is as follows-

$$wt_i^{'} = Ctf_i * \log\left(\frac{D^{'}}{df_i}\right) \quad \text{Equation -2}$$

Here Ctf_i denotes total term frequency or number of times a term occurs in a document vector D'
D' denotes subset of document vector D.
df_i denotes document frequency or number of documents snippet consists term t_i

It is observed in many cases that various terms have the similar weights. Therefore it is not feasible to select the terms for query expansion simply by calculating their weight and selecting the highest weight tem. Another aspect of the query expansion is the context of the terms in relation to query. The context of the terms is calculated by calculating the distance of the query terms with document key terms. To calculate the average distance of the highest weighted terms with the query terms, we had used the following formula, where –

dt_i denotes the average distance of the highest weighted term i in the document vector D'

d_i denotes distance of the highest weighted term i in the nth document vector D'

n denotes total number of documents of document vector D' which consists of the highest weighted term

$$dt_i = \sum_{i=1}^{n} \frac{d_i}{n} \qquad \text{Equation - 3}$$

To calculate the average distance of the term we had declared x size window around the query terms if an individual term falls in this window then the closest distant term will assigned distance weight 2, next will assigned 1.5 and if the term falls at the critical size of the window then 1. The average distance is calculated by the Equation – 2.

The final weight is calculated by the Equation - 3

wt_i denotes final weight of the weighted term i

$wt_i^{'}$ denotes initial weight of the highest weighted term i

dt_i denotes the average distance of the highest weighted term i in the document vector D'

$$wt_i = wt_i^{'} + dt_i \qquad \text{Equation - 4}$$

The term with the highest weight is selected for the query expansion. The new expanded query is given to the search engines and the relevancy is sorted out by calculating the precision and the similarity score of the results. We get the result set with the improved precision and similarity score.

6.3 Performance comparison of Query Expansion on three Search Engines

The performance of WSD and query expansion approach is evaluated on the basis of the results of three popular Hindi language search engines. Table 1 shows the original P@10 value for the three search engines against the five queries. Table 2 shows the P@10 values for the expanded queries.

Table 1. P@10 values for the five queries in the three search engines

Query#	P@10 on Google	P@10 on Raftaar	P@10 on Guruji
सोना और स्वास्थ्य	0.23	0.20	0.10
गुलाब की **कलम**	0.40	0.50	0.20
दण्ड बैठक	0.44	0.30	0.30
अंक विज्ञान	0.50	0.40	0.20
कर्ण प्रिय संगीत	0.40	0.40	0.10

Table 2. P@10 values for the five queries after WSD and query expansion in the three search engines

Query#	P@10 on Google	P@10 on Raftaar	P@10 on Guruji
निद्रा **सोना** और स्वास्थ्य	0.90	0.90	0.40
पौधे गुलाब की **कलम**	1.0	0.80	0.30
व्यायाम **दण्ड** बैठक	1.0	0.20	0.10
ज्योतिष **अंक** विज्ञान	1.0	0.70	0.60
मधुर **कर्ण** प्रिय संगीत	1.0	0.80	0.60

7. Discussion

With the advent of the IT revolution now native language users also look for the information on web. Besides technical problems in Hindi language web IR, sense ambiguity is one of the major issues for Hindi language web IR. According to the goal of this paper we observed the performance level of the three popular search engines for Hindi language web information retrieval.

In Table 1 the precision values with the original queries in the three search engines are shown. It is observed that precision values for the ambiguous queries are quiet low. However in comparison to Google and Raftaar P@10 values for Guruji is very low.

On evaluation of P@10 values for the expanded queries it has been observed that the performance of Google is improved very much in comparison to Raftaar and Guruji. Raftaar also shows improvement. However in Guruji result improvement is very low.

8. Conclusion

Hindi language search engines are facilitating the users but only to some extent. The results after the sense disambiguation and expansion are compared for the three search engines Google, Raftaar and Guruji. The results show an overall increase of precision in all the three search engines. However if compared individually the highest improvement of precision is in Google, Raftaar shows an average increase in performance and Guruji shows the lowest increase in performance. The close observation identified that the performance of Guruji was also not very

good with the original queries. In some cases it is found that the performance of Guruji deteriorates after disambiguation and query expansion. Hence we can conclude that the maximum benefit of the approach is in the case of Google and before disambiguation and query expansion its performance was better than the other two search engines.

References

[1] I. Klapaftis and S. Manandhar, "Google & WordNet based Word Sense Disambiguation", in Proceedings of the Workshop on Learning and Extending Ontologies by using Machine Learning methods, Bonn, Germany, 2005

[2] M. Á Gaona, R., Gelbukh, A. and S.Bandyopadhyay, "Web-based variant of the Lesk approach to Word Sense Disambiguation", in Proceedings of Eighth Mexican International Conference on Artificial Intelligence, Guanajuato, Mexico, 2009, pp. 103-107.

[3] R Navigli,. and G. Crisafulli, "Inducing Word Senses to Improve Web Search Result Clustering". in Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP 2010), MIT Stata Center, Massachusetts, USA, 2010, pp. 116-126.

[4] F. Pinto and C. Sanjulián, "Automatic query expansion and word sense disambiguation with long and short queries using WordNet under vector model". SISTEDES, 2008, pp. 17-23.

[5] J. Nie and F. Jin, "Integrating Logical Operators In Query Expansion In Vector Space Model", in Proceedings of ACM SIGIR-2002 Workshop on Mathematical and Formal Methods in Information Retrieval. Tampere, Findland, 2002, pp. 77-88.

[6] R. Mandala and T. Tokunaga, "Combining multiple evidence from different types of thesaurus for query expansion". in Proceedings of ACM-SIGIR'99, 1999, pp. 191-197.

[7] E. M. Voorhees, "Using WordNet to disambiguate word senses for text retrieval", in Proceedings of ACM-SIGIR'93, Pittsburgh, 1993, pp. 171-180.

[8] S.K.M Wong, W. Ziarko and P.C.N. Wong, "Generalized vector space model in information retrieval", in Proceedings of ACM-SIGIR'95, 1985, 18-2

[9] S. Dwivedi and P. Rastogi, "An Entropy Based Method for Removing Web Query Ambiguity in Hindi Language", Journal of Computer Science. Vol. 4, No. 9, 2008, pp. 762-767.

[10] R. Baeza-Yates and B, Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, 1999

[11] J. Rocchio, "Relevance feedback in information retrieval", in G. Salton (Ed.), The smart retrieval system experiments in automatic document processing Englewood Cliffs, NJ Prentice-Hall, 1971, pp. 313 -323.

[12] T. Nykiel and H. Rybinski, "Word Sense Discovery for Web Information Retrieval", in Proceedings of the 2008 IEEE International Conference on Data Mining Workshops, 2008 pp. 267-274.

Author Biographies

Parul Rastogi has obtained her MCA Degree in the year 2003 from IGNOU. Her research interests are Web Mining, Word Sense Disambiguation and Hindi language Information Retrieval. She is pursuing Ph.D. in Computer Science. Ms. Parul Rastogi has published many of the valuable research papers in various national and international journals. She is member of Computer Society of India (CSI).

Dr. S.K. Dwivedi has obtained his Ph.D. Degree from Banasthali Vidyapeeth in the year 2006. He has completed his Ph.D in the area of Web Mining. His research interests are Web content Mining, Semantic Web, Search Engine performance evaluation etc. He has published many of the valuable research papers in various national and international journals. He is presently working as a Associate Professor of Computer Science dept, of BBAU, Lucknow, India.