Optimizing Search Results using Wikipedia based ESS and Enhanced TF-IDF Approach

Amit Rajeshwarkar
Government Engineering College,
Aurangabad, INDIA

Meghana Nagori
Government Engineering College,
Aurangabad, INDIA

ABSTRACT

The Triangular Search approach aims at recalculating authenticity of the Search Results provided by the Google API with the help of Semantic similarity provided by Wikipedia API and calculating the cosine similarity between the Document Vectors and query string vectors using enhanced approach of Tf-Idf that reduces calculation involved in it.

The Search Engine Optimization traces anchor texts that are the values between <a> tag of HTML and body texts of a web page. Using the Vector Space Model, the Term frequency and Inverse document frequency are calculated along with the Page ranking algorithm to get the Search Results. But consideration of anchor texts in search engine optimization techniques leads to some of the non-relevant body texts of a document. Also the top results of a search engine include trending and e-commerce links other than sponsored links but the intent of search is not considered.

This approach proposes and gains user intents behind the search thereby focusing on providing intent related search results.

General Terms

Optimized Search Results

Abbreviations

ESS/ ESA: Explicit Semantic Similarity/Analysis
TF-IDF : Term Frequency Inverse Domain Frequency

API : Application Programmable Interface

Keywords

Google API, Wikipedia API, Explicit Semantic Analysis (ESA), Enhanced TF-IDF.

1. INTRODUCTION

Search Engines have been a major part of people’s lives, they act as the librarian who finds a book from multiple stacks of books with different sections, very similar to Search engines that help extract the webpages that a browser intends to find. But for the librarian to do so, he needs to understand the book needed by the reader that can be analogous to understanding the context of the search in order to provide accurate results this can be done with the help of the dis-ambiguities provided by Wikipedia API. Thus focused crawlers assist to narrow down the search and get the more accurate results [1]. Now in order to say a book corresponds to users request the librarian needs to know the content of the book in abstract and whether the user intended work can be included in the book, so he forms an algorithm to maintain list of books and the metadata of the book, in the model being used, the tf-idf (Term Frequency Inverse Domain Frequency) algorithm helps us calculate the occurrence of keywords from the search query in the Search results from search engines and the aggregate of cosine similarities of query terms with link titles(anchor texts) and body texts of a page. The greater is the value of cosine similarity aggregate, the more is the document relevant to the entered search query [2].

The Search results from search engines are to be processed, these can be extracted using Google URL API i.e. http://ajax.googleapis.com/ajax/services/search/web?v=1.0&q =YourSearchStringHere When connected to the above link by appending with it, the entered search query, a json (Java Script Object Notation) page that has all the results relevant to the query as per Google algorithm are retrieved.Wikipedia API i.e.

http://en.wikipedia.org/w/api.php?action=query&titles=Searc hString can be used to find semantically similar terms for a entered query. The ambiguities related to search query can be removed by prompting user to select from the different contexts of the queries. For example an apple may resemble to a fruit family or a Company or some other ambiguity [3].

This paper is organized in Six sections, Section 1 deals with Introduction Section 2 gives The Related Study that includes the existing infrastructures of TF-IDF and Semantic Search Algorithms, Section 3 briefs the Implementation of existing Algorithms and Section 4 shows System Analysis and Section 5 concludes the paper whereas section 6 specifies the references.

2. RELATED WORK

The approach is to provide a categorized view of Search results, where the largest knowledge hub Wikipedia specifies some information about the search and narrow down the search by removing dis-ambiguities and remaining results from engine are processed using an enhanced approach of tf-idf. Results are prioritized links relevant to the search term.

2.1 Vector Space Model

The term frequency inverse domain frequency is calculated by following formula for all terms present in document that are of some significance, while other words such as helping verbs, pronouns are known as stop words e.g. he, she, it, then, is, was, have, etc. which are removed.

$$\text{RelScore}_d = \sum_{t \in q \cap d} w_{t_d}, w_{t_d} \equiv TF_{t,d} \cdot IDF_t$$

The term frequencies of significant words are calculated by considering its occurrences in the texts related to a links body text. Where,

Term frequency TF (t,d) = count of query term ‘t’ in body text of a Link document.

The count however may be very huge as there may be infinite numbers of links on web and many of them contain term t in ample. So in order to ease up calculation, all the term frequencies of a document are divided up by the greatest value amongst them. After having normalized TF table, their values lie between 0 and 1.

And Inverse Document Frequency (IDF) of term t is given by

IDF (t) = Log10 (N/c)

Where c is number of documents in which term t is present and N is total number of documents. Now taking the product of Normalized TF scores of each significant word for each document and IDF of keywords gives the TFIDF vector of documents [4].

The same method is used to calculate TFIDF vector for query string. The resulting vector is query vector.

Cosine similarity relates the query with most relevant document, the cosine value of 0 is 1 and cosine value of 90 is 0.Similarly when the cosine similarity is calculated by

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2}\sqrt{\sum\limits_{i=1}^{n} B_i^2}}$$

It represents the inclination of the document vector towards the query vector. Greater is the cosine similarity values, greater is its relevance.

2.2 Page Ranking

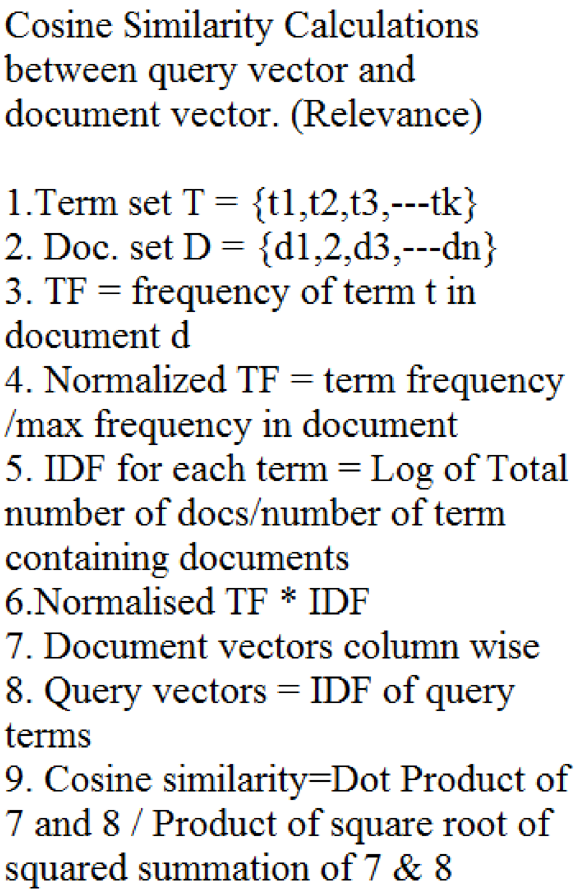
Page ranking algorithms helps Search engines optimize results by ranking pages according to the citations they get. That is when a page has some quality contents it is referenced by many, thus that page gets preference compared to non-frequently referred page [5].

2.3 Explicit Semantic Similarity

Wikipedia being the largest and quality source of information is used in our system to find out semantically similar terms related to a search string. Wiki-Relate! can also be considered but when there are no articles with same search string in Wiki-Relate, it fails[6]. So Wikipedia ESA, which can figure out articles and hence semantically similar search terms related to a search string, even if the articles with same name does not exist in its database, qualifies it for our use.

2.4 Enhanced TF IDF

Removal of only stop words and keeping all the significant words increases tedious calculations. Thus in enhanced tf-idf approach only words of query string along with some semantic similar keywords such as hyper-nyms, hyponyms, synonyms, antonyms, alias names and other such replacements for query words are to be taken in consideration thus instead of calculating document vector for all significant words in a document only query words and its semantic words vectors are calculated [7].



Cosine Similarity Calculations between query vector and document vector. (Relevance)

1.Term set T = {t1,t2,t3,---tk}
2. Doc. set D = {d1,2,d3,---dn}
3. TF = frequency of term t in document d
4. Normalized TF = term frequency /max frequency in document
5. IDF for each term = Log of Total number of docs/number of term containing documents
6.Normalised TF * IDF
7. Document vectors column wise
8. Query vectors = IDF of query terms
9. Cosine similarity=Dot Product of 7 and 8 / Product of square root of squared summation of 7 & 8

Fig. 1: Summary of Calculation of Cosine similarity between Query Vectors and Documents Vectors

3. SYSTEM IMPLEMENTATION

The steps included in implementing this paper are as follows

1. The input query is searched using existing search engines like Google using Google APIs to get their top priority links. (For Traditional Search)
2. All Wikipedia articles related to the search string are extracted from Web using JSOUP parser or existing APIs like media-Wiki for Wikipedia.
3. The semantically similar keywords are taken into account to calculate semantic relatedness of a webpage to the search string (using Vector space model).
4. TFIDF vectors for each document is checked against TFIDF vectors of search string including semantically similar terms in search string
5. Cosine similarity value represents the relevance wise ranking for the web pages with the entered search string and semantically related keywords which are extracted from Wikipedia.

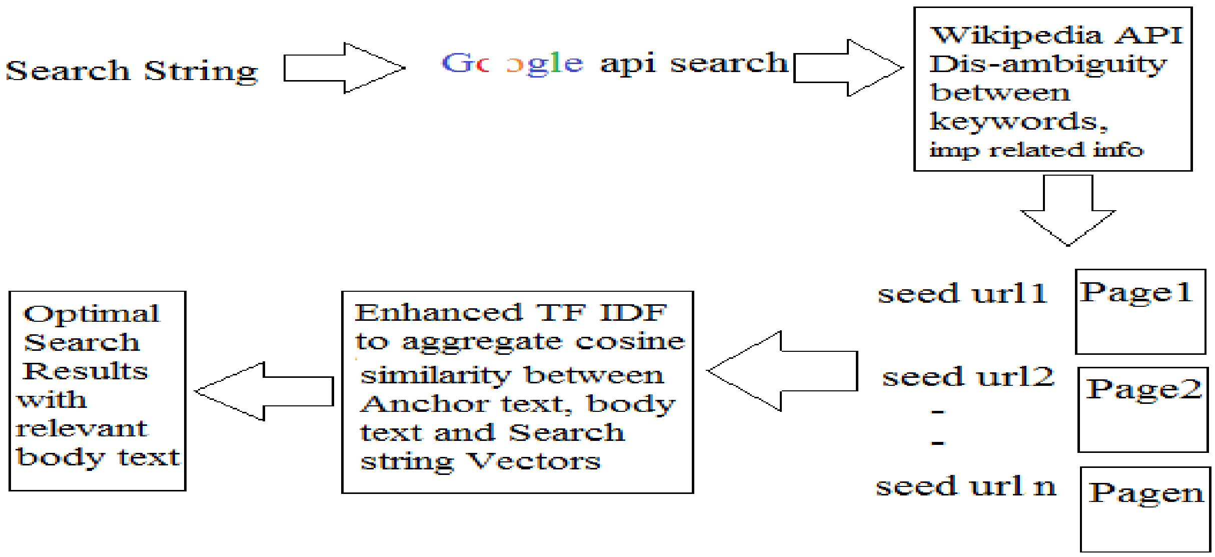
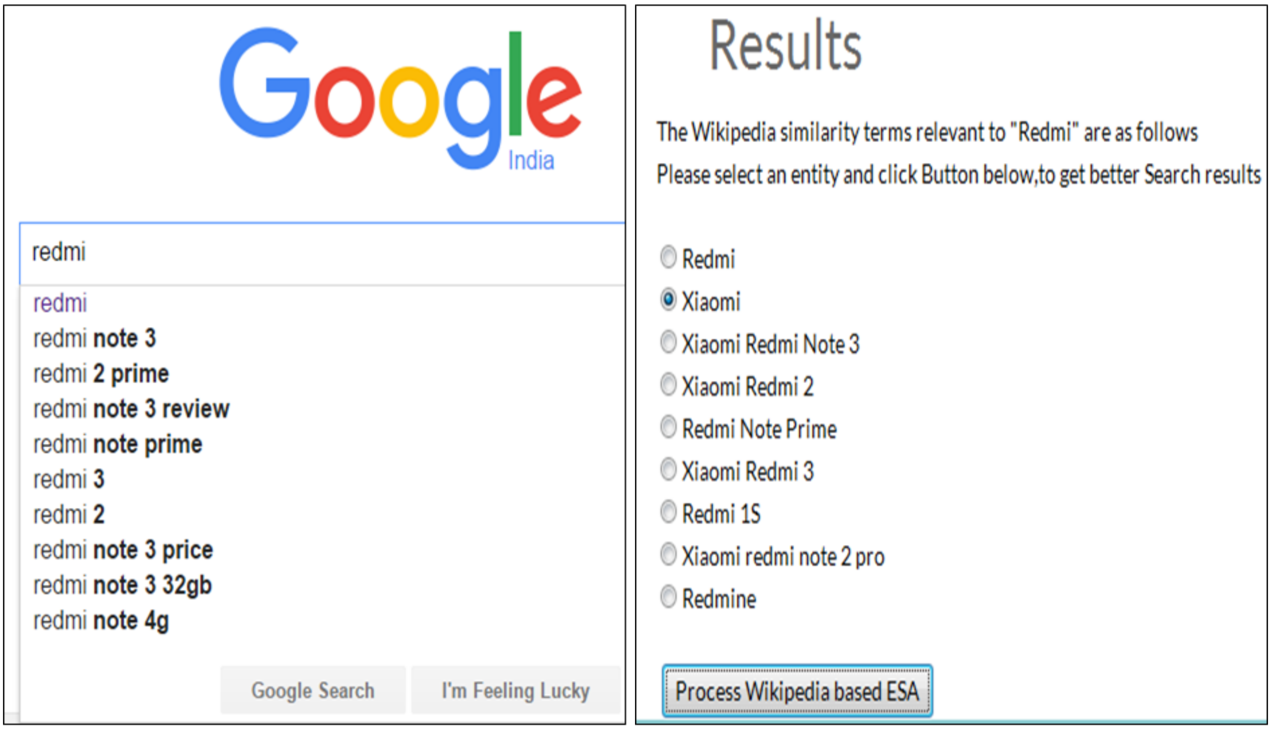

Search String
Gc ɔgle api search
Wikipedia API Dis-ambiguity between keywords, imp related info
seed url1
Page1
seed url2
Page2
-
-
seed url n
Pagen
Enhanced TF IDF to aggregate cosine similarity between Anchor text, body text and Search string Vectors
Optimal Search Results with relevant body text

Fig.2: Proposed System to attain better search results with disambiguation using Wikipedia and removal of hoax due to SEO by searching in anchor as well as body text.



redmi

redmi
redmi **note 3**
redmi **2 prime**
redmi **note 3 review**
redmi **note prime**
redmi **3**
redmi **2**
redmi **note 3 price**
redmi **note 3 32gb**
redmi **note 4g**

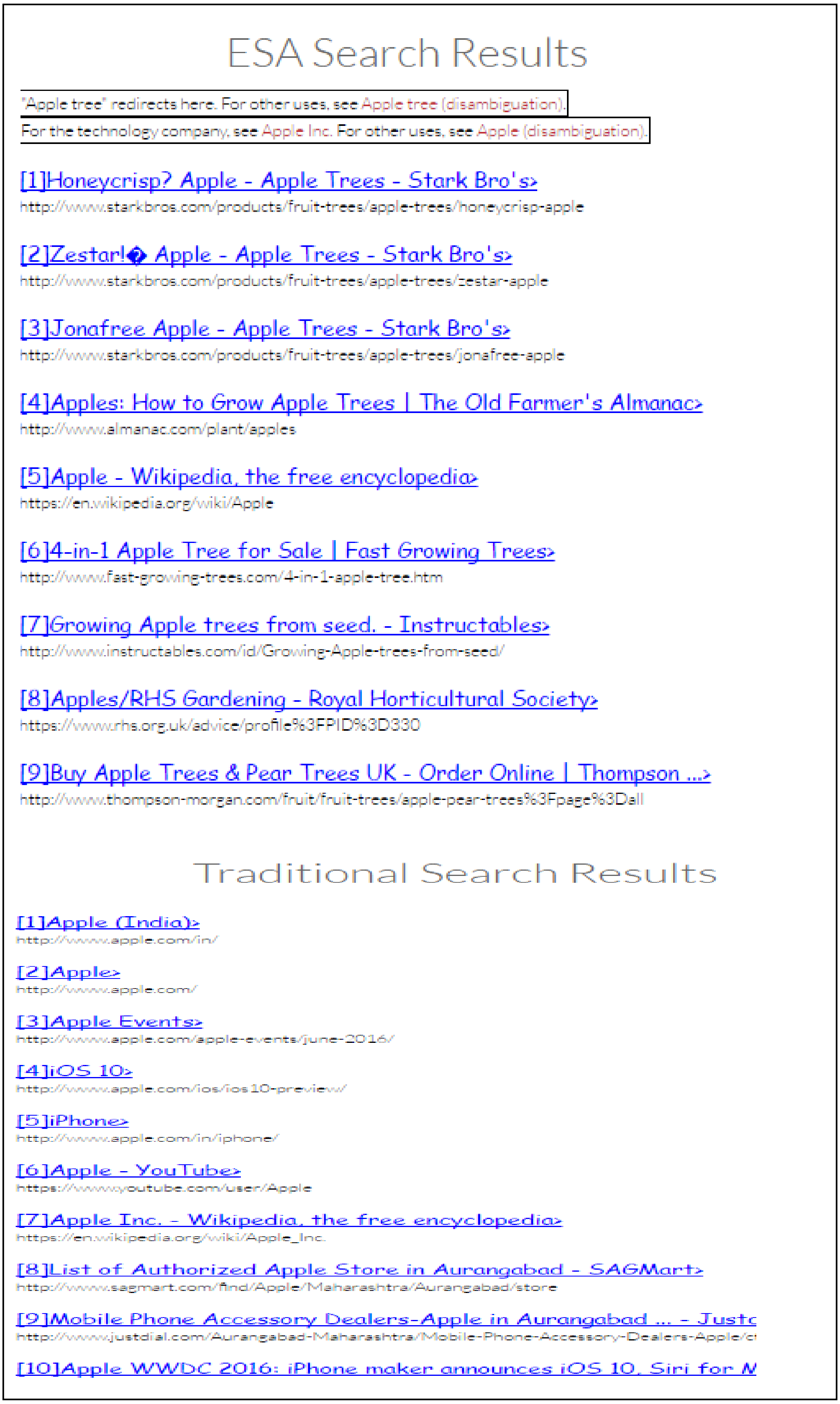
Results

The Wikipedia similarity terms relevant to "Redmi" are as follows

Please select an entity and click Button below,to get better Search results

○ Redmi
● Xiaomi
○ Xiaomi Redmi Note 3
○ Xiaomi Redmi 2
○ Redmi Note Prime
○ Xiaomi Redmi 3
○ Redmi 1S
○ Xiaomi redmi note 2 pro
○ Redmine

Fig 3: Existing - Only Lexical Search Suggestions Vs. Proposed - Semantically similar Search suggestions



ESA Search Results

"Apple tree" redirects here. For other uses, see Apple tree (disambiguation).

For the technology company, see Apple Inc. For other uses, see Apple (disambiguation).

[1]Honeycrisp? Apple - Apple Trees - Stark Bro's>
http://www.starkbros.com/products/fruit-trees/apple-trees/honeycrisp-apple

[2]Zestar!� Apple - Apple Trees - Stark Bro's>
http://www.starkbros.com/products/fruit-trees/apple-trees/zestar-apple

[3]Jonafree Apple - Apple Trees - Stark Bro's>
http://www.starkbros.com/products/fruit-trees/apple-trees/jonafree-apple

[4]Apples: How to Grow Apple Trees | The Old Farmer's Almanac>
http://www.almanac.com/plant/apples

[5]Apple - Wikipedia, the free encyclopedia>
https://en.wikipedia.org/wiki/Apple

[6]4-in-1 Apple Tree for Sale | Fast Growing Trees>
http://www.fast-growing-trees.com/4-in-1-apple-tree.htm

[7]Growing Apple trees from seed. - Instructables>
http://www.instructables.com/id/Growing-Apple-trees-from-seed/

[8]Apples/RHS Gardening - Royal Horticultural Society>
https://www.rhs.org.uk/advice/profile%3FPID%3D330

[9]Buy Apple Trees & Pear Trees UK - Order Online | Thompson ...>
http://www.thompson-morgan.com/fruit/fruit-trees/apple-pear-trees%3Fpage%3Dall

Traditional Search Results

[1]Apple (India)>
http://www.apple.com/in/

[2]Apple>
http://www.apple.com/

[3]Apple Events>
http://www.apple.com/apple-events/june-2016/

[4]iOS 10>
http://www.apple.com/ios/ios10-preview/

[5]iPhone>
http://www.apple.com/in/iphone/

[6]Apple - YouTube>
https://www.youtube.com/user/Apple

[7]Apple Inc. - Wikipedia, the free encyclopedia>
https://en.wikipedia.org/wiki/Apple_Inc.

[8]List of Authorized Apple Store in Aurangabad - SAGMart>
http://www.sagmart.com/find/Apple/Maharashtra/Aurangabad/store

[9]Mobile Phone Accessory Dealers-Apple in Aurangabad ... - Justc
http://www.justdial.com/Aurangabad-Maharashtra/Mobile-Phone-Accessory-Dealers-Apple/ct

[10]Apple WWDC 2016: iPhone maker announces iOS 10, Siri for M

Fig 4: Comparison between top k Results from a search engine before and after model is implemented for intent "animal" and term "apple"

4. ANALYSIS AND GRAPHS

4.1 Analysis of Tools/Mechanisms used.

4.1.1 ESA Wikipedia vs. other tools

The tool, best suited for both single word or/and phrase/text relatedness is ESA-Wikipedia from the below analysis. [6]

Table 1: Computing word and text relatedness using different tools.

Tools	Word relatedness	Text relatedness
Word-Net	0.33–0.35	-
Bag of Words	-	0.1-0.5
Roget's Thesaurus	0.55	-
LSA	0.56	0.6
Wiki-Relate	0.19-0.48	-
ESA-Wikipedia	0.75	0.72
ESA-ODP	-	0.69

4.12 Calculations reduced: Enhanced TF IDF

Table 2: Calculations reduced to significantly (where T : Total Significant terms , t : Query terms , d : # of documents and T is significantly greater than t)

Steps	TF-IDF Approach (Calculations)	Enhanced TF-IDF (Calculations)
TF	T * d	(t) *d
Normalized TF	T * d	(t) *d
TF-IDF	T * d + (T)	(t) *d + (t)
Cosine Similarity	T * d	(t) *d
Total Calculations	4(T*d) + T	4(t) *d + (t)
Calculations Reduced	-	4(T-t) *d + (T-t)

4.2 Model Evaluation: Statistical Analysis

Manual intervention for semantic analysis is made with 5 judges and their average is considered. A study proves Human based semantic relatedness is consistently higher than any of the machine learning tools. [8]

The Analysis is made using Precision and Recall which is given by

4.2.1 Precision

$$\text{Precision} = \frac{|\{\text{relevant documents}\}\cap\{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

The Search term being changed 5 times and analyzed by 5 judges, the average is shown in below table

Table 3: Precision Table calculated with different search engines

Semantic Search Engines	No. of Relevant Documents/ Total retrieved	Precision
DuckDuckGo	22/210	0.11
SenseBot	6/25	0.24
Rengine	19/70	0.27
Triangle	9/10	0.9

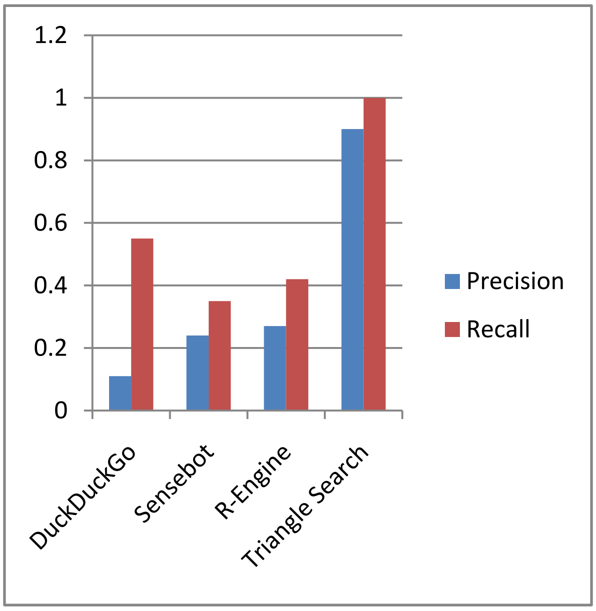
4.2.2 Recall

$$\text{Recall} = \frac{|\{\text{relevant documents}\}\cap\{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

The same scheme of averaging results of 5 judges for 5 different terms is used for calculating recall

Table 4: Recall Table calculated with different search engines

Semantic Search Engines	No. of Relevant Retrievals/ Total Relevant	Recall
DuckDuckGo	22/40	0.55
SenseBot	6/17	0.35
Rengine	19/45	0.42
Triangle	9/9	1



Graph 1: Results Comparison with other Search Engines

5 CONCLUSION

The Triangle Search Approach works in three dimensions to improve the search results using enhanced tf-idf approach that improves efficiency by reducing calculations significantly still providing the same results, secondly with the help of explicit semantic source such as Wikipedia, to gain insight into context of query and hence suggesting the possible intents search query may relate to and lastly to return narrowed results according to the user context selection.

Such highly improved precision and recall values are account of an intermediate step to predict and gain intent of the search. Search engine optimization considers anchor texts to find similarity, but sometimes links may misguide, also the top results of popular search engines include only trending and sponsored or ecommerce links, but this approach provides results to the point and most of the times the partially known keyword may also help redirect to the most appropriate results.

Future work may include use of Big data analytics such as flume to process data on a greater platform so that larger data be processed and gain even more better search results that may have been left out.

6 REFERENCES

[1] Yajun Du, Wenjun Liu, et al. 2015. “An improved focused crawler based on Semantic Similarity Vector Space Model”. The Official Journal of the World Federation on Soft Computing (WFSC) 36, Elsevier, 392–407.

[2] What Is Search Engine Optimization / SEO. 2011. Common Craft, Search Engine Land. YouTube. We. 12 Sep.2011.

[3] Masumi Shirakawa, Kotaro Nakayama, et al.2015 “Wikipedia based Semantic Similarity Measurements for Noisy Short Texts Using Extended Naive Bayes”. IEEE Transactions on Emerging Topics in Computing, DOI 10.1109/TETC.2418716.

[4] Z. Yun-tao, et al., 2005. “An improved TF-IDF approach for text classification”. Journal of Zhejiang University SCIENCE, 6A(1):49-55.

[5] H. A. Haddadene, et al., 2012.”On the Pagerank Algorithm for the Articles Ranking”. Proceedings of the World Congress on Engineering, Vol I July 4 - 6, 2012.

[6] E.Gabrilovich et al., 2007. “Computing semantic relatedness using Wikipedia based Explicit Semantic Analysis”. IJCAI, 1606-1611.

[7] A.Hliaoutakis, G.Varelas, et.al.,2006 “Information retrieval by semantic similarity”, I. J. Semant. WebInf. Syst. 3 (3) 55–73.

[8] Budanitsky, Hirst et al. 2006. “Evaluating WordNet based Measures of Lexical Semantic Relate-dness”. Computational Linguistics: ACM Digital Lib., 32(1), 13-47.