International Journal on Future Revolution in Computer Science & Communication Engineering
Volume: 4 Issue: 4
ISSN: 2454-4248
207 – 210

A Systematic Review on Stopword Removal Algorithms

Jashanjot Kaur
Department of Computer Science and Engineering
Sant Longowal Institute of Engineering and Technology
Longowal, Sangrur (Punjab) – 148106, India
jashanjot211993@gmail.com

Preetpal Kaur Buttar
Assistant Professor
Department of Computer Science and Engineering
Sant Longowal Institute of Engineering and Technology
Longowal, Sangrur (Punjab) – 148106, India
preetpalkaur15@gmail.com

***Abstract*—** Stopwords, also known as noise words, are the words that contain a little information which is not usually required. Stopwords were discovered by H.P. Luhn in 1958. In the domain of information retrieval, an effective indexing can be achieved by removing the stopwords. Indexing is a technique of connecting or tagging documents with different search terms or criteria. The main motive behind the elimination of stopwords is to increase the execution speed and the accuracy. It not only decreases the vector space but also helps to improve overall performance. It also helps in reducing the size of text. Till now, techniques for automatic stopwords removal have been developed for languages such as English, Sanskrit, Arabic, Chinese, etc. In this paper, we discuss the different techniques which have been used by the researchers to construct automated stopword lists in different languages.

Keywords- *Stopwords; noise words; stopword list.*

*****

I. INTRODUCTION

The current subject of significant global research includes text classification, document clustering, and similar document analysis tasks since such areas support the enterprises of web intelligence, web mining, web search engine design, and so forth. An important aspect of all the machine learning tasks involving document processing is a list of 'stopwords', also known as 'stoplist'. The documents involved in information retrieval tasks contain a huge proportion of such data that are not useful for the researchers. Therefore, it is desired that some automatic method should be developed to identify such data and to remove it from dataset before its processing. Such data are known as stopwords. Stopwords were first introduced in 1958 by H.P. Luhn. Stopwords are the words that occur most frequently in a document and contain a little information that is not usually required. For example, in English language, there are some words such as a, about, above, after, again, against, all, am, an, and, any, are, as, at, be, been, could, do, does, during, etc. that occur most frequently in a text, but contain meagre information. These words are called stopwords. A set of stopwords is known as 'stopword list' or 'stopword corpus'.

Removing stopwords does not only reduce the vector space but also improve the performance by increasing the execution speed, calculations, and also the accuracy. For example, if your search query in the context of a search engine is "how to develop android app". The search engine then tries to find web pages that contain the terms "how", "to" "develop", "android", "app". The search engine finds more pages that contain the terms "how" and "to" than the pages containing information about developing information retrieval applications because the terms "how" and "to" are most commonly used terms in the English language. So, if these two terms are disregarded, the search engine can focus on retrieving pages that contain the keywords: "develop", "android", "app", that will result in bringing up the pages that are really of interest. These words are removed in preprocessing phase of the text classification process which helps in reducing the size of text. Stopwords can also be removed manually but it is a time-consuming process which is proportional to the corpus size.

II. PROPERTIES OF STOPWORDS

Two facts were discovered by Luhn in the field of information retrieval[1]. Firstly, a relatively small number of words account for a very significant fraction of all text's size. Words like 'it', 'and' and 'to' can be found in virtually every sentence in English-based documents. Secondly, these words make very poor index terms[2]. These words have low discrimination value and the information carried by these words is negligible. These words are also known as noise words or negative dictionary that appear frequently in documents and does not carry a useful information to aid learning tasks. Therefore, these types of poor words lead to the degradation in accuracies. The properties of stopwords can be summarized as below:

- Stopwords are the words with low discrimination power.
- The specific nouns, verbs or other grammatical types could be having less candidature for being stopwords and the elements like articles, prepositions, and conjunctions are usually present in a stopword list.
- Stopwords serve only a syntactic function and never have any predictive capability. They do not indicate the subject matter.
- They have a very high frequency so they can affect the efficiency of the information retrieval process.
- They affect the weighting process as stopwords are tend to diminish the impact of frequency differences between less common words.
- The document length can be changed by the removal of the stopwords and affects the weighting process.
- The fact that if they carry no meaning, they can also affect the efficiency, resulting in a large amount of unproductive processing.

III. TYPES OF STOPWORDS

Stopwords are generally a single set of words. It means different for different types of applications. For example, a stopword list can contain:

- **Determiners:** the, a, an, another
- **Coordinating conjunctions:** for, an, nor, but, or, yet, so

 ISSN: 2454-4248
207 – 210

- **Prepositions:** in, under, towards, before

But in domain-specific cases, for example, in clinical texts, a different set of stopwords is required, for example, “mcg”, “dr”, and “patient” that may have low discriminating power in constructing intelligent applications compared to terms such as “heart”, “failure”, and “diabetes”.

IV. METHODS OF STOPWORD REMOVAL

Following are the most commonly used techniques for the removal of stopwords from a text:

The Classic Method: This method is used to remove stopwords obtained from pre compiled lists.

Methods based on Zipf’s Law (Z-Methods): Three stopword creation methods are used in addition to the classic stoplist. This includes removing most frequent words (TF-High), removing words that occur once, i.e., singleton words (TF1), and removing words with low inverse document frequency (IDF).

The Mutual Information Method (MI): It is a supervised method that is used by computing the mutual information between a given term and a document class (e.g., positive, negative), providing a solution of how much information the term can tell about a given class. Low mutual information suggests that the term has a low discrimination power and it should be removed consequently.

Term based Random Sampling(TBRS): This method was first proposed by Lo et al.[3] in which the stopwords are detected manually from web documents. This method is used by iterating over randomly selected separate chunks of data and ranks terms in each chunk based on their in-format values using the Kullback-Leibler divergence measure as shown in the following equation:

$$d_x(t) = Px(t).\log_2 Px(t)/P(t)$$

where,

Px(t) is the normalized term frequency of a term t within a mass x, and

P(t) is the normalized term frequency of t in the entire collection.

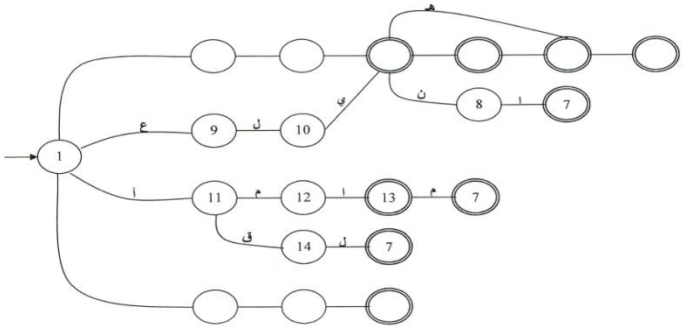
The final stop list is constructed by taking the least informative terms in all chunks by removing all possible duplications.

V. LITERATURE SURVEY

Many researchers have done undertaken the task of stopword removal by suggesting some statistical methods for finding the stopwords. A lesser amount of work has been done in other languages comparatively to the research done in English language.

Sinka & Corne, 2003[4] have proposed a new stopword list based on word-entropy. They introduced optimization of a stoplist and stochastic search combining k-means clustering which results in better performance. The new stoplists are derived on the basis of information theoretic measures that are calculated on Bank Search dataset[4] and a random dataset. The authors have performed set of experiments and on the basis of their research, they found that the median performance turned out as 93.05% of Van Rijsbergen stoplist[4]. The results of two sets of ten trial runs each are reported, one for hillclimbing (HC) and other for evolutionary algorithm (EA). The accuracies of the stoplists in hillclimbing experiment on the classification task are 95.35% and 96.1%. The accuracies of HC have been improved upon Van Rijsbergen stoplist by 2.35% and 3.1%. The performance of EA is better than HC method, but it has slightly lower mean performance than HC. The EA accuracy is 95.58%, whereas, HC is 95.66%. But they achieved the best result with accuracy of 96.15% as compared with Van Rijsbergen stoplist with the accuracy of 93% and achieved best overall result (96.2%) with an improvement of 3.2 points in accuracy. Therefore, all experiments achieved about atleast 1.8% improvement upon Van Rijsbergen list.

Al-shalabi et al., 2004[5] proposed and implemented an efficient stopword removal algorithm for Arabic language based on a Finite State Machine(FSM), as some stopword list techniques based on the use of a dictionary are expensive and time-consuming. Moreover, these techniques require more space to store data. About 242 Arabic abstracts were chosen and more than 1000 Arabic stopwords were taken from the proceedings of the Saudi Arabian National Computer conferences with 47897 words to test new stopword removal technique for Arabic Language. Another set of data was taken from the holy Quraan. The authors used deterministic Finite Automata(DFA) that accepts a word find out whether a word is a stopword or not as shown in the figure given below:



The authors ran the system on 550 MHz PC. The system took 26 seconds and resulted in 12891 stopwords. About 7030 Arabic words were taken from the holy Quraan and resulted in 3235 stopwords. The accuracy of a system was 98%.

Lo et al., 2005[3] proposed different methods in building a stopword list automatically for an information retrieval system. A new approach has been used called term- based sampling by using Kullback-Leibler divergence measure and assess the result by using four TREC collections. Term-based sampling approach is compared to various approaches based on Zipf’s law and determines whether a word contains useful information. The baseline approaches are applied such as term-frequency, normalized term frequency, inverse document frequency and normalized IDF. The aim of using these approaches is to produce best average precision.

El-Khair, 2006[6] investigated three stopwords list and their effectiveness for Arabic information retrieval. The stopword lists the authors used is the general stoplist, the corpus-based stoplist, and the combined stoplist. The inverse document frequency weight, probabilistic weighting, and statistical language modelling are the three popular weighting schemes that were examined by them. The aim is to improve the performance and compare the effect of combined statistical approaches with linguistic approaches. The data set used with the Lemur Toolkit is LDC (Linguistic Data Consortium). A general stoplist resulted in better overall performance than the other two lists. Wilcoxon test and recall and precision curves were used to compare results. About 6 different techniques with 12 different combinations were examined. The test statistic $\chi 2 = 70.471$ and the P-value*= 0.000 indicates that the differences that are determined by the Friedman two-way ANOVA test are statistically significant. Therefore, the Kullback-Leibler divergence model had some problems when performed with stopwords. The characteristics of the best match algorithm, BM25 weight, can be combined with language specific characteristics to enhance further improvements in order to develop new weighting algorithm.

Zou et al., 2006[7] proposed an automatic aggregated methodology for statistical and information models to extract Chinese stopword list in order to save the time and to extract the stopword list manually. The result that is analysed shows that the Chinese stopword list made

 ISSN: 2454-4248
207 – 210

by the authors is compared with English stop word list but it is more general as compared to other Chinesestopword lists. A stopword extraction algorithm is proposed which could also be applied to other languages in the future. These words were extracted from TREC 5 and 6 Chinese corpora which was widely accepted as standard corpora for Chinese processing. This corpus contains news reports from both Xinhua newspaper and People's daily newspaper. Stopwords are extracted by statistical model on the basis of probability and distribution. The authors have eliminated non-Chinese symbols in the preprocessing step. The two lists are aggregated together to generate the final one. Stopwords are classified into two categories. First is "generic stopwords" which are the stopwords in the generic domain. The other type is "document-dependent stopwords", also known as "domain stopwords". The words like "Britain" and "govern" in the Zipf list are not included in most generic stopword list, because they are domain stopwords of TIME magazine.

Alhadidi & Alwedyan, 2008[8] developed a stopword removal technique for Arabic language. A set of 242 Arabic abstracts has been taken from the proceedings of the Saudi Arabian National Computer conferences and another set of data is chosen from the Jordanian AlraiNewspaper. Approximately 92% of the stopwords are removed by using this technique. It is a hybrid technique as the stopwords are removed based on the dictionary and the algorithm. Two lists were compared and the same results were achieved.

Dolamic & Savoy, 2009[9] evaluated the two stopword lists for the English language (one comprising 571 words and another with 9 words) and then compared them with a search approach accounting for all word forms. Mean Average Precision(MAP) is used as a retrieval-effectiveness measure that changes the performance of the stopword list when compared, but without the removal of stopwords. For English, a shorter stop word list containing 9 words gives the same performance as the longer stop word list of 571 words and for French language also similar results are obtained. With Hindi and Persian languages, long stopword list is compared with an indexing strategy which also leads to a significant improvement. Moreover, some implementations were also done on the traditional Okapi IR model and DFR paradigm which lead to low retrieval-performance levels.

R.Puri et al., 2013[10] proposed a suitable and automated method for the stopwords identification in Punjabi Language. A stopword list was obtained by searching the most frequent words in the document using a statistical method. The probability of words from the documents was analyzed in another approach and the result showed the possibility of a word being a stopword. These two approaches were used to find the aggregated stopword list. The authors took 10,000 news articles from Punjabi newspaper "Ajit". The average of the words taken was 400 words per article. All the special characters, digits and punctuation marks were eliminated. A stopword list was generated from the frequency count as well as frequency distribution of the words. After sorting the list, the words were ranked in descending order as per their position. Then the final score was obtained by performing the sum of the ranks of each word. The stopwords with higher probability are the words which are at the top of the list. The authors concluded that the aggregation of the two lists can change the order of some words in the original lists.

Garg, & Goyal, 2014[11] created a Hindi stopword list based on the frequency of words in the documents.. The percentage of stopwords in any document was calculated and experimentally analyzed. The authors discussed the similarity of two documents that contain Hindi text. The experiments performed by the authors suggest that removal of stopwords decreases the similarity of Hindi text documents as the similarity score rises due to the presence of frequent words. Therefore, it was concluded that removing the stopwords decreases the degree of comprehension of the text, but removal of stopwords is important for more accuracy of information retrieval tasks.

Raulji et al., 2016[12] used a simple approach to design stopword removal algorithm and its implementation for Sanskrit language. It is a hybrid approach that was used for the creation of a generic stopword list containing 75 stopwords. The algorithm and its implementation used dictionary-based approach. In dictionary-based approach, a predefined list of stopwords is compared to the target text on which removal is required. The algorithm that was implemented was tested on about 2 MB of data that contains nearly 87,000 Sanskrit words collected from web and other digital media, out of which nearly 11,200 stopwords were eliminated. The stopwords were eliminated from 6 different documented texts. Total number of words in the text were reduced by approximately 13% which also reduced CPU cycles for data processing. The accuracy obtained was approximately 98%. The authors reported that scarcity of digitized availability of written textis an issue with Sanskrit language.

Siddiqi & Sharan, 2017[13] constructed a generic list of stopwords for Hindi language without corpus statistics with the help of linguistic experts, e.g., "और", "का", "के", etc. This stop word list contains more than 800 stop words. But the list lacks in quantity and quality of stop words due to unavailability of its inflected stopwords. For the completeness of the list, the authors have added inflected variants of a particular stop word, e.g., a stop word "उन" is followed are the words which are its inflected variants, such as, "उनका", उनकी","उनके", "उन्हें", "उनसे", "उनको", "उनमें", etc. The common words are added to the domain-specific words.

VI. COMMON STOPWORD LISTS

A number of stopword lists based on these methods are available on the internet. these lists have also been adopted as standard stopword lists in many research works. the following table lists some the freely available stopword lists.

TABLE I. SOME OF THE FREELY AVAILABLE STOPWORD LISTS

Stopword list	Language	URL
Snowball stopword list	English	http://snowball.tartarus.org/
Terrier stopword list	English	http://bitbucket.org/kganes2/text-mining-resources/downloads/terrier-stop.txt
Indian Language Technology Proliferation and Deployment Centre Stopword List	Hindi	https://github.com/stopwords-iso/stopwords-hi/blob/master/stopwords-hi.txt
Minimal Stopword list	English	https://bitbucket.org/kganes2/text-mining-resources/downloads/minimal-stop.txt
Forum for information retrieval evaluation	Hindi	https://www.isical.ac.in/~fire/data/stopwords_list_hin.txt

ISSN: 2454-4248
207 – 210

CONCLUSION

Stopwords are the no-information words which do not contribute any information to the text-processing task at hand. Instead, they may reduce the accuracy and degrade the performance if included in the text processing because they only a small number of stopwords constitute a large fraction to the total text. It is therefore required to eliminate the stopwords during the preprocessing phase. A number of stopword removal methods have been developed by the researchers in the past, particularly for the English language. There is a requirement of efficient stopword removal techniques to be developed for other languages also.

REFERENCES

[1] H. P. Luhn, "A Statistical Approach to Mechanized Encoding and Searching of Literary Information," *IBM J. Res. Dev.*, vol. 1, no. 4, pp. 309–317, 1957.

[2] R. K. Belew, "Finding Out About: A Cognitive Perspective on Search Engine Technology and the World Wide Web," *Uma ética para quantos?*, vol. XXXIII, no. 2, pp. 81–87, 2000.

[3] R. T.-W. Lo, B. He, and I. Ounis, "Automatically Building a Stopword List for an Information Retrieval System," *J. Digit. Inf. Manag. Spec. Issue 5th Dutch-Belgian Inf. Retr. Work.*, vol. 5, pp. 17–24, 2005.

[4] M. P. Sinka and D. W. Corne, "Evolving Better Stoplists for Document Clustering and Web Intelligence," *Des. Appl. hybrid Intell. Syst.*, pp. 1015–1023, 2003.

[5] Riyal al-shalabi, ghassan kanaan, jihad M. jaam, Ahmad Hasan, and Eyad Hilat "Stop-Word Removal Algorithm for Arabic Language.Pdf," *Information and Communication*. pp. 1–5, 2004.

[6] El-Khair, "Effects of stop words elimination for Arabic information retrieval: a comparative study," *Int. J. Comput. Inf. ...*, vol. 4, no. 3, pp. 119–133, 2006.

[7] F. Zou, F. Wang, X. Deng, and S. Han, "Evaluation of Stop Word Lists in Chinese Language," *5th Ed. Int. Conf. Lang. Resour. Eval.*, pp. 2504–2507, 2006.

[8] B. Alhadidi and M. Alwedyan, "Hybrid Stop-Word Removal Technique for Arabic Language.," *Egypt Comput Sci*, vol. 30(1), no. 1, pp. 35–38, 2008.

[9] L. Dolamic and J. Savoy, "When Stopword Lists Make the Difference," *Am. Soc. Inf. Sci. Technol.*, no. 1, pp. 200–203, 2009.

[10] R. Puri, R. P. S. Bedi, and V. Goyal, "Automated Stopwords Identification in Punjabi Documents," *An Int. J. Eng. Sci.*, vol. 8, no. June 2013, pp. 119–125, 2013.

[11] U. Garg and V. Goyal, "Effect of Stop Word Removal on Document Similarity for Hindi Text," *An Int. Jounal Eng. Sci.*, vol. 2, no. December, 2014.

[12] J. K. Raulji, R. Scholar, B. Ambedkar, J. R. Saini, I. / C. Director, and R. Supervisor, "Stop-Word Removal Algorithm and its Implementation for Sanskrit Language," *Int. J. Comput. Appl.*, vol. 150, no. 2, pp. 975–8887, 2016.

[13] S. Siddiqi and A. Sharan, "Construction of a generic stopwords list for Hindi language without corpus statistics," *Int. J. Adv. Comput. Res.*, vol. 8, no. 34, pp. 35–40, 2017.