Stopword Identification and Removal Techniques on TC and IR applications: A Survey

Dhara J. Ladani
M.Tech (IT) Student
Department of Information Technology
Dharmsinh Desai University
Nadiad, India
dharaladani@gmail.com

Nikita P. Desai
Department of Information Technology
Dharmsinh Desai University
Nadiad, India
npd.ddit@gmail.com

***Abstract*—The concept of "Stopword" was first introduced by H.P. Luhn in 1958. In Natural Language Processing (NLP), Stop word is a common word that is neither indexed nor searchable in a computer search engine. Example of stop words are 'a', 'the', 'is' etc. Removing stopword is Pre-processing step in majority of NLP applications, including IR (Information Retrieval) and TC (Text Classification). Some of the benefits of removing stop word are - decrease in size of corpus by 35-45%, improvement of efficiency and accuracy of the text mining applications thus helping in reduction of time and space complexity of overall application. In this paper, we discuss the various major stopword identification techniques used by the researchers in last few decades, for Indian Language and Non-Indian Languages. Also, we present a survey of methods used for stopword list generation with their characteristics. We have also mentioned the effect of various stopword removal techniques applied on TC and IR application domains. A comprehensive list of resources publicly available for static stop words in various languages is also given for quick reference.**

Keywords—stopword, Information Retrieval, Text Classification, Bag-of-words, Noisy Word, Text Mining, Natural Language Processing, Indian Language

I. INTRODUCTION

Nowadays Natural Language Processing (NLP) is one of the most recent topics for research which had applications like Information Retrieval (IR), Text Classification (TC), Document Clustering, Sentiment Analysis (SA), Question-Answering etc. Reasons for it being in focus are– it supports web intelligence companies, web mining, web search engine design, and so on. A stopword is an important aspect of all machine learning tasks involving document processing. The stopword concept was first presented in 1958 by H. P. Luhn. As per the British dictionary, stopwords are common word that is not indexed or searchable in a computer search engine. For example, in English language, few stopwords are 'a', 'and', 'the', 'as', 'an', 'all', 'do', etc. In the context of IR application, search engine ignores stopword and only focuses on retrieving pages that contain the important keywords which would bring up pages that are actually of interest. Thus, removing stopword not only reduces the text corpus size but also increases the processing power, reduces index entries for searching, and subsequently space and time complexity improves [22]. In the TC application, where the text is to be classified into different categories, if stopword is excluded from the given text more focus can be given to those words which define the meaning of the text [22].

II. BRIEF IDEA ABOUT STOPWORD

In this section, we discuss stopword properties, types, benefits of removing stopword, and its supportive and non supportive applications. It might help beginners to understand the concept of stopword.

A. Properties of stopword

- Stopword is a word with low discrimination power. And it is only used for connecting high discriminating power words while building sentences. For example, in English language 'a', 'an', 'the', 'and', etc.
- It has no meaning.
- It never has any predictive capability.
- It has a high frequency of occurrence in the text.

B. Types of stopword

Stopwords are broadly categorized into two groups-

1) Generic stopword: Generic stopword cover routine words which are language-specific and general words used in all areas [16]. For English language, example are- 'a', 'and', 'the', 'all', 'do', 'so', 'an', etc.

2) Domain-Specific stopword: It covers a particular domain-specific word. The domain could be like education, medical, sports, bollywood, politics and many more [16]. Example -For Education domain: 'pen', 'table', 'student', 'book, 'classroom', 'board', etc.

C. Benefits of removing stopword

- It reduces text count in the document corpus by nearly 35-45% [7].
- By removing the stopword, the size of the dataset decreases. Also, the time to train and test the model decreases [13].

978-1-7281-5197-7/20/$31.00 ©2020 IEEE

- Removing stopword can also help to improve the performance of IR and TC.
- The proficiency and exactness of the text mining application improves [22].

D. When to remove stopwords

Removing stopword is a good idea for the applications like information retrieval (IR), auto-tag generation, caption generation and for text classification tasks like spam filtering, language classification, etc.

E. When to not remove stopwords

Removing stopword is not an good idea while we are performing the task such as a machine translation, language modelling, text summarization question-answering problems, sentiment analysis, etc. [6]. For example, in SA if we consider "not" as a stopword, then removing it might lead to misinterpretation in the review.

III. Background Knowledge

In this section we discuss different types of stopword identification techniques. The techniques are further grouped as static and dynamic approach based methods.

A. Static Approach Based stopword Identification

In a static approach, a predefined stopword list is created for the particular language. The list itself would not need to be refreshed or changed automatically.

1) Classic Method: This is a basic technique [21] in which targeted documents are tokenized. It uses a static list for stopword to identify the tokens of document. List is made manually.

2) Deterministic Finite Automata: It creates a finite automata [7,1] for listed stopwords as in fig- 1. DFA consists of five parameters that are state, character, transition, start state and accepting state. Starting conditions are tested for each character and transitions are made to next state. If no transitions exist, false is returned. If subsequence transitions on input symbols of the word end in accepting state, the word is considered stopword & it returns true.

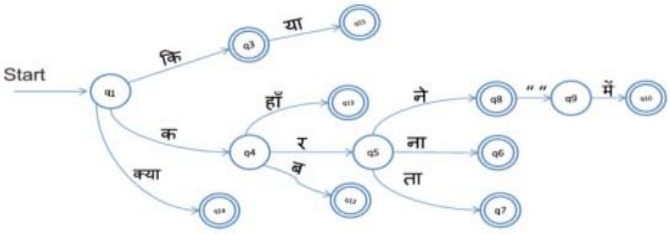

Fig.1. DFA State Diagram [7]

3) Lexical Class Method: In this method [6], with the assistance of phonetic specialists and profound learning of Gujarati language structure, it creates unique words list and assigns lexical classes to them i.e. Noun, Verb, Adverb, etc. In the next step, it counts the frequency of highest frequency words as per lexical classes.

B. Dynamic Approach Based Stop-word identification

In a dynamic approach, the stopword is identified on the go. A stopword list will be generated as per rules or statistics & it is not fixed apriorly. The list of stopword is decided based on given text source.

1) Rule-based Approach Method: In this technique [8] static rules are derived after analysis of the pattern of stopword. If any word in the text is satisfying the rule then it is identified as a stopword.

2) Methods based on Zipf's Law: Zipf's law [16] is a law on frequency of word distribution in a language. To illustrate Zipf's law let us suppose we have a collection of text and let there be V (vocabulary) unique words in the collection. For each word in the collection we need to compare the frequency i.e. Freq(word)= The total occurrence of the term in a document. Then we assign rank to the words in descending order by their frequency (most frequency word has rank 1, next frequency word has rank 2, and so on. Zipf's law expresses that "Given a large sample of words used, the frequency of any word is inversely proportional to its rank in the frequency. The following are some more computations done [16].

a) Term Frequency (TF): The total occurrence of the term happens in a particular document.

$$TF= t/doc(t) \quad (1)$$

where, T= total occurrence of the term "t" appears in a document, doc(t)= total number of the terms in the document.

b) Normalized Term Frequency (NTF): NTF is created by normalizing the term frequency by the total number of tokens in the collection i.e. the size of the dictionary document is given by,

$$TFNorm=-\log(TF/V) \quad (2)$$

where, TF = Term Frequency, V = Total number of tokens in the dictionary document.

c) Inverse Document Frequency (IDF): It is a measure of how much information the word provides.

$$IDF(k)= \log(NDoc/Dk) \quad (3)$$

where, NDoc = Total number of documents in the corpus, Dk = Number of documents containing the term "k".

d) Term Frequency Inverse Document Frequency (TF-IDF): It works on concept that if word is common across all documents, it is not having much discriminating power. It is calculated by,

$$TF\text{-}IDF=TF*IDF \quad (4)$$

3) Term Based Random Sampling (TBRS): This approach is detecting stopword manually from web document. Utilizing the Kullback-Leibler difference measure as iterates over randomly selected separate chunks of the data and rank in-format terms in each chunk dependent on their in-format values.

$$dx(t)= Px(t).\log 2Px(t)/P(t) \quad (5)$$

where, Px(t) = NTF of a term "t" within a mass "x" P(t) = NTF of "t" throughout the collection. The final stopword list is

generated by eliminating all potential duplication and after taking the least detailed words in all pieces.

IV. SURVEY OF STOPWORD IDENTIFICATION TECHNIQUES

In this section we discuss various stopword identification technique for different languages.

A. Stopword Identification Techniques

Many researchers have worked on stopword identification techniques. They have applied various feature extraction techniques. For Non-Indian language most of the researchers have used Zipf's law for automatically identifying stopword and have achieved acceptable accuracy.

Work by AI-Shalabi et al. [1] is based on Finite State Machine for Arabic language. The main goal was reducing the problem of using the dictionary-based approach which is more time and space consuming task. They achieved accuracy 98%.

Yao et al. [3] have created a 1289 Chinese stopword list by merging the classical stopword list with the different domain stopword.

Rakholia et al. [6] for Gujarati language created a list of unique 190 words with the aid of a linguistic expert and deep learning of Gujarati grammar.

Jha et al. [7] developed an algorithm based on deterministic finite automata (DFA) to extract stopwords from the Hindi text. The algorithm is tested on 200 documents and was successful with an accuracy of 99%. They also claim it takes 1.77 second whereas dictionary-based takes 3.4 seconds.

Rakholia et al. [8] proposed a rule-based approach & dynamically identify stopwords for Gujarati language. They have created static 11 rules and applied them for creating a stopword list at runtime. They claim to have achieved accuracy for generic stopword of 98.10% and Domain-specific 94.08%.

Ayral et al. [14] have proposed an automated method for generating domain-specific stopword to improve the classification problem. Here researchers modified the bayesian characteristic language classifier, which relied on a posteriori probability estimation of appropriations using the bag-of-words model to check the generated stopwords and checked method on website pages.

Raulji et al. [21] used a dictionary-based approach which has 75 generic stopword. The implemented algorithm was tested on about 2MB of data containing almost 87,000 Sanskrit words obtained from the internet and other digital media, out of it almost 11,200 stopwords were removed. The total number of words in the text was decreased by about 13% which also reduced CPU cycles for data processing. The accuracy achieved was about 98%.

Table I shows the concise analysis of corpus source, size of dataset and result achieved during implementation for various stopword identification techniques evolved during the last few decades.

TABLE I. SURVEY OF STOPWORD IDENTIFICATION TECHNIQUE

Researchers	Target Language	Technique	Approach	Corpus source and size	Testing Dataset	Result	Accuracy (in %)
AI-Shalabi et al. (2004) [1]	Arabic	DFA	Static	Created more than 1,000 words using Bonnie Glover Stalls and Yaser AI-Onaizam, and Another from translating English to Arabic	242 Arabic abstracts from SANCC with Total 47894 words	12891 stopwords in SANCC	98%
					7030 words From Holy Quraan of total 7030 Arabic words	3235 stop words in Holy Quraan Dataset	
Zou et al. (2006) [2]	Chinese	Aggregation of Statistical and information model	Dynamic	423 English articles in TIME magazine (total no. of words is 245,412)	Chinese TREC 5& 6	Generated stopword list is more general than another list and also it comparable with English stopword list	N/A
Yao et al. (2008) [3]	Chinese-English	Sequence Filter, MRU Filter, Hash Filter	Static	FUDAN University 204 documents in Chinese with English abstract	N/A	Customized 1289 Chinese stopword list	N/A
						Based on time consuming Hash-Filter algorithms best	
Alajmi et al. (2012) [4]	Arabic	TF, Mean & variance, Entropy, Aggregation	Dynamic	A corpus of 1002 documents (each document contains 700,000 words in which 140781 are unique word)	N/A	Top 200 stopwords	96%
Victor et al. (2013) [5]	Yoruba	Entropy based	Dynamic	Two sets of corpora of 756,039 Yoruba	N/A	List of 256 stopwords from diacritized and removing it reduces full	N/A

				words for diacritized and its undiacritized version		text by 65.91% List of 189 stopwords from undiacritized & removing it reduce full text by 67.46%	
Rakholia et al. (2016) [6]	Gujarati	Lexical Classes Based	Static	Routine Gujarati news and articles content 126 text document where each document contained nearest 260 tokens	N/A	Discovered top 190 stopwords that as often as possible utilized in practically all Gujarati written documents.	N/A
Jha et al. (2016) [7]	Hindi	DFA	Static	200 documents from http://hindi.webdumia .com/bollywoodmove -review	Hindi documents from EMILLE corpus	It takes 1.77 -sec v/s dictionary-based take 3.4 sec	99%
Rakholia et al. (2017) [8]	Gujarati	Rule-based	Dynamic	373 Gujarati documents (each document contain near 400 words) 224 documents for domain specific i.e. Medical& Engineering (each document contains near 275 words)	N/A	N/A	98.10% for routine document 94.08% for domain specific document
Siddiqi et al. (2017) [9]	Hindi	Own algorithm	Dynamic	N/A	N/A	Create a generic stopword list of 800+ words	98.10% for routine document
Rani et al. (2018) [10]	Hindi	Statistical model Knowledge based (Shannon Term Entropy) Technique	Dynamic	various online news portals such as "NaiDunia", "Vigyandunia", "Center for Advanced Study of India", "Kisanhelp" etc. (11,800+ total no. of documents)	N/A	Top 70 stopword found	N/A
Rani et al. (2018) [11]	Hindi	Statistical model Information Theory (Entropy) Model	Dynamic	newspaper (total no. of documents is 554)	Editorial articles from NaiDunia	Compare constructed stopword list with GitHub stoplist 1 GitHub stoplist 2 Rank stoplist Siteskevin stoplist	98.3 % 97. % 97.% 92.%
Miretie et al. (2018) [12]	Amharic	Aggregate of TF, IDF, Entropy value measure	Dynamic	From magazines, newspapers and blogs	N/A	The proposed method outcome the problem of using dictionary based which are inefficient, very expensive and time-consuming	N/A
Ayral et al. (2011) [14]	English	Bag of words, Bayesian Maximum Posteriori Classifier	Dynamic	PASCAL ontology dataset	N/A	Top 20 words generated by using the Proposed method, other list generated by term occurred in unique documents, by unique topics and based on the Oxford English corpus	N/A

B. Survey on stopword removal effect on TC and IR Domain

In this section we discuss the usefulness of stopword removing technique in TC and IR domains. Also, various statistical models & corpus details are discussed for each.

Silva et al. [13] used Reuter-21578 corpus and split it as 75% for training and 25% for testing purposes. They used preprocessing step like-low frequency word remove, stopword removal, and stemming. They conclude that stopword provides very minimal information for the task of classifying text. Removing stop words may substantially reduce the size of the text function space and help speed up the calculation and improve the accuracy of text classification.

Gunasekara et al. [16] created database from online newspaper on different domains. They applied the statistical stopword removal technique on Sinhala news classification and checked average F-measure and average accuracy for the

various types of stopword list. They used MaxEnt and NB classifiers.

To improve the Mean Average Precision (MAP), Joshi et al. [18] proposed the stopword elimination process for Gujarati IR domain. They gathered corpus from FIRE for the experiment and created a list covering 400 words that are less significant and often used in the language of Gujarati. With the aid of linguistic experts and study, they produced 282 stopword.

Table II shows the survey of stopword identification and removal, done on TC & IR applications.

TABLE II. SURVEY OF STOPWORD IDENTIFICATION AND REMOVAL IMPORTANCE ON TC & IR DOMAINS

Researchers	Target Language	Technique	Classifier/ Model	Corpus source and size	Testing Dataset	Result
Silva et al. (2003) [13]	English	Bag- of- word	SVM	http://kdd.ics.uci.edu/databases/ reuters21578/reuters21578.html.R e uters-21578 is a financial corpus covers 12000 classified stories into 118 categories	25% (3299 items) from collected data corpus	For removing stopword, low freq. words and stemming cover 97% Accuracy, 85.91% precision, 62.54% Recall, 71.77% F1
Shama et al. (2015) [15]	General	Term weighting	N/A	Used Google web API to collect the 64 documents on the Ipad concept it has 9998 features	N/A	Removal of stopwords decreases the size of feature space sparsity value 0.9 as 90%.
Gunasekara et al. (2018) [16]	Sinhala	TF, NTF, IDF, NIDF, TFDN, TF*IDF	NB, MaxEnt	Online newspapers (www.dinamina.com, www.silumina.lk, www.lankadeepa.lk), www.ucsc.cmb.ac.lk/ltrl& SinHG5 corpus that covers 1000 documents with 5 categories	N/A	MaxEnt is more sensitive to remove stopword. Without stopword NIDF achieve 99.74% accuracy
Pandey et al. (2009) [17]	Hindi	N/A	VSM Model	N/A	EMILLE corpus no. of documents are 700 & no. of query are 70.	Create 350 generic stopword list After removing stopword 45.04% MAP, 22.45% reduce index size and also give better result in retrieval effectiveness.
Joshi et al. (2013) [18]	Gujarati	N/A	Mean Average Precision (MAP)	daily newspaper "Gujarat Samachar" (2001-2010) & size is 2.7 GB cover 3,13,163 no. of documents	Fire 2011	After elimination of stopword improve title as 3.6%, combination of title and description as 8% and combination of title, description, and narration as 24%.
Atwan et al. (2013) [19]	Arabic	N/A	VSM Model	LDC Arabic Newswire dataset with 383,872 documents contain 76 million tokens over nearest 666,094 unique word	TREC 2001 & 2002	Light stemmer with combined stopword list increase 9% data reduction rate
Rakholia et al. (2018) [20]	Gujarati	Zipf's law	N/A	Gujarati website 272 documents (each document contains almost 4800 words and total no. of words are 132,509,08 & unique words are 53,566	N/A	To improve the performance of IR only medium TF will be consider instead of head and tail frequent term

C. *Analysis of some of available stopword list & comparison between available stopword identification techniques in Indian Languages*

Here we discuss some available stopword list. These lists are also used in many applications as standard stopword list. Also analyze some of the advantage and disadvantage of stopword identification techniques for Indian Languages.

Table III shows the details about the name of the organization, target language, link where the stopword list exist and total how many stopwords they cover.

Table IV shows the details of the different types of stopword identification techniques implemented by researchers on different language with their advantages and disadvantages.

TABLE III. AVAILBLE STOPWORD LIST

Stopword Organization	Target Language	URL of list	No. of stopwords
Indian Language Technology Proliferation and Deployment Centre	Gujarati	https://github.com/gujarati-ir/Gujarati-Stop-Words	91

Stopword List			
Indian Language Technology Proliferation and Deployment Centre Stopword List	Hindi	https://github.com/stopwords-iso/stopwords-hi/blob/master/stopwords-hi.txt	225
Forum for information retrieval evaluation	Hindi	https://www.isical.ac.in/~fire/data/stopword_list_hin.txt	97
Indian Language Technology Proliferation and Deployment Centre Stopword List	English	https://gist.github.com/sebleier/554280	127
Minimal Stopword list	English	https://bitbucket.org/kganes2/textminingresources/downloads/minimal-stop.txt	85
Terrier stopword list	English	http://bitbucket.org/kganes2/text-mining-resources/downloads/terrier-stop.txt	733
Indian Language Technology Proliferation and Deployment Centre Stopword List	Arabic	https://github.com/mohataher/arabic-stop-words/blob/master/list.txt	750
Arabic stopword list	Arabic	https://www.ranks.nl/stopword/arabic	104

TABLE IV. COMPARISION OF STOPWORD IDENTIFICATION TECHNIQUES

References	Technique	Implemented on Language	Advantages	Disadvantages
Saini et al. [21]	Classic Approach	Sanskrit	● Basic Technique ● Easy to implement	● Lacks potentially new words ● Defined for general purpose i.e. different collection require stopword list ● Outdated ● Time complexity is high
Rakholia et al. [6]	Lexical Classes Approach	Gujarati	Best for Machine Translation	As of date no tool available for Gujarati language to assign automatic POS to word
Jha et al. [7]	Deterministic Finite Automata	Hindi	Take less time as compare to dictionary-based approach	● Limited word length ● Require more space to store data
Rakholia et al. [8]	Rule-based Approach	Gujarati	Dynamic	This algorithm work based on created rule hence, it cannot handle neologism.

V. CONCLUSION

In this paper, detailed survey results are shown to highlight issues of different stopword identification techniques. Also available dataset, various classifier techniques and models for identifying the importance of removing stopword in TC and IR application were studied. From this survey, we conclude that removing stopword reduces the corpus size, improves precision, recall and accuracy value and also reduces the space and time complexity for searching/ indexing. For generic and domain-specific stopword identification techniques, there are mainly two methodology - Static and Dynamic. Only a few researchers have worked on dynamic approach. Especially for domain-specific stopwords, dynamic method of generation is not applied. The static approach won't be able to handle "new" stopwords; as list is predefined. Whereas dynamic approach would take more time in identification of stopword but can handle OOV - out of vocabulary words also. As per our survey we observed that many researchers have done work for Non-Indian languages like English, Arabic, etc. but very few work has been done for Indian languages like Gujarati.

References

[1] R. Al-Shalabi, G. Kanaan, J. M. Jaam, A . Hasnah, and E. Hilat, "Stop-word removal algorithm for Arabic language," InProceedings. 2004 International Conference on Information and Communication Technologies: From Theory to Applications, 2004 Apr 23, p. 545, IEEE.

[2] F. Zou, F. L. Wang, X. Deng, S. Han, and L. S. Wang, "Automatic construction of Chinese stop word list," InProceedings of the 5th WSEAS international conference on Applied computer science, 2006 Apr 16, pp. 1010-1015.

[3] M. Makrehchi, and M. S. Kamel, "Automatic extraction of domain-specific stopwords from labeled documents," InEuropean Conference on Information Retrieval 2008 Mar 30, pp. 222-233, Springer, Berlin, Heidelberg.

[4] A. Alajmi, E. M. Saad, and R. R. Darwish, "Toward an ARABIC stop-words list generation," International Journal of Computer Applications. 2012 May, pp. 8-13.

[5] T. V. Asubiaro, "Entropy-based generic stopwords list for Yoruba texts," International Journal of Computer and Information Technology. 2013 Sep, 2(5).

[6] R. M. Rakholia, and J. R. Saini, "Lexical classes based stop words categorization for Gujarati language," In 2016 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA), pp. 1-5, IEEE.

[7] V. Jha, N. Manjunath, P. D. Shenoy, and K. R. Venugopal, "Hsra: Hindi stopword removal algorithm," In2016 International Conference on Microelectronics, Computing and Communications (MicroCom), 2016 Jan 23, pp. 1-5, IEEE.

[8] R. M. Rakholia, and J. R. Saini, "A Rule-Based Approach to Identify Stop Words for Gujarati Language," In Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications, 2017, pp. 797-806, Springer, Singapore.

[9] S. Siddiqi, and A. Sharan, "Construction of a generic stopwords list for Hindi language without corpus statistics," International Journal of Advanced Computer Research, 2018, 8(34), pp.35-40.

[10] R. Rani, and D. K. Lobiyal, "Social Choice Theory Based Domain Specific Hindi Stop Words List Construction and Its Application in Text Mining," InInternational Conference on Intelligent Human Computer Interaction, 2018 Dec 7, pp. 123-135, Springer, Cham.

[11] R. Rani, and D. K. Lobiyal, "Automatic Construction of Generic Stop Words List for Hindi Text," Procedia computer science, 2018 Jan 1, 132:3, pp.62-70.

[12] S. G. Miretie, and V. Khedkar, "Automatic Generation of Stopwords in the Amharic Text," International Journal of Computer Applications, 975:8887.

[13] C. Silva, and B. Ribeiro, "The importance of stop word removal on recall values in text categorization," In Proceedings of the International Joint Conference on Neural Networks, 2003 Jul 20, Vol. 3, pp. 1661-166, IEEE.

[14] H. Ayral, and S. Yavuz, "An automated domain specific stop word generation method for natural language text classification," In 2011 International Symposium on Innovations in Intelligent Systems and Applications, 2011 Jun 15, pp. 500-503, IEEE.

[15] D. Sharma, and S. Jain, "Evaluation of stemming and stop word techniques on text classification problem," International Journal of Scientific Research in Computer Science and Engineering, 2015, 3(2), pp. 1-4.

[16] S. V. Gunasekara, and P. S. Haddela, "Context aware stopwords for Sinhala Text classification," In2018 National Information Technology Conference (NITC), 2018 Oct 2, pp. 1-6, IEEE.

[17] A. K. Pandey, and T. J. Siddiqui, "Evaluating effect of stemming and stop-word removal on hindi text retrieval," InProceedings of the First International Conference on Intelligent Human Computer Interaction, 2009, pp. 316-326, Springer, New Delhi.

[18] H.Joshi, J. Pareek, R. Patel, and K. Chauhan, "To stop or not to stop—Experiments on stopword elimination for information retrieval of Gujarati text documents," In2012 Nirma University International Conference on Engineering (NUiCONE), 2012 Dec 6, pp. 1-4, IEEE.

[19] R. M. Rakholia, and J. R. Saini, "Information Retrieval for Gujarati Language Using Cosine Similarity Based Vector Space Model," InProceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications, 2017, pp. 1-9, Springer, Singapore.

[20] R. M. Rakholia, and J. R. Saini. "Impact of zipf's Law in Information Retrieval for Gujarati Language" International Archive of Applied Sciences and Technology, 2018.

[21] J. K. Raulji, and J. R. Saini, "Stop-word removal algorithm and its implementation for Sanskrit language," International Journal of Computer Applications. 2016 Sep;150(2):15-7.

[22] J. Kaur and P. K. Buttar. "A systematic Review on Stopword Removal Algorithms," International Journal on Future Revolution in Computer Science & Communication Engineering, pp. 207-210.