IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.7, Special Issue.1, Jan 2019 E-ISSN: 2347-2693

Algorithm for Removal of Semantically Insignificant Content Words

Abhijit Barman[1*], Diganta Saha[2]

[1,2]Departmet of Computer science and Engineering, Jadavpur University, Kolkata, India

[*]*Corresponding Author: abhijitbarmancob@gmail.com, Tel.: +91-94745-18993*

Available online at: www.ijcseonline.org

***Abstract*—** This paper describes how the context specific semantically insignificant content words are extracted using Inverse Document Frequency (IDF) and Inverse Class Frequency (ICF) measure. We are able to remove around 42% of total corpus volume as irrelevant information which includes textual noise, function words and context specific semantically insignificant content words. We have executed different Machine Learning(ML) algorithms used for text classification on a corpus, before and after the removal of the textual noise. We found that there have been no significant change in accuracy of those ML algorithms before and after removal of the textual noise.

***Keywords*—** Machine Learning(ML), Natural Language Processing(NLP), Information Retrieval (IR), Term Document Matrix, Inverse Document Frequency (IDF) and Inverse Class Frequency (ICF), Stop Words, Content Words.

I. INTRODUCTION

In the era of digitization, the quantity and availability of digital text documents is increasing day by day giving raise of demand of a number of NLP applications like Machine Translation, Information Retrial, Text Summarization, etc. Processing of huge amount of text is resource intensive and time consuming and often error prone due to noise and irrelevant information present in the digital text data. So it is required to prepossess the digital text documents to remove noise and irrelevant information before actual computaion. Typically millions of` unique terms present in a corpus thus the feature space is huge which in turn affects the performance and accuracy[1][2]. It is important to extract and remove the words that have no or low semantically significance. For example, function words which are present in a sentence to explain syntactic relationship among words present in the sentence and have no semantic significance. Also there are many contents words present in a corpus which are insignificant in a particular context or domain. In the proposed approach we are able to remove around 42% of total corpus volume as irrelevant information which includes textual noise, function words and context specific semantically insignificant content words.

Rest of the paper is organized as follows, Section II contain the related work of removal of textual noise or insignificant words, Section III contain assumption and notations used in the algorithm, section IV explain the proposed algorithm with steps, Section V describes results and discussion of the outcome of the algorithm, VI concludes research work with future directions.

II. RELATED WORK

A Stop word or Function word present across the all documents in a corpus with high frequency. So, such a word do not carry any significant meaning or semantic property. Sinka & Corne, 2003[3] proposed a word-entropy based stop word. Al-Shalabi et al., 2004[4] had designed Finite State Machine (FSM) to eliminate stop-words for Arabic Language. Alhadidi & Alwedyan, 2008[5] developed hybrid a stop word removal technique using both dictionary and algorithm for Arabic language. Dolamic and Savoy, 2009[2] had shown the effect of stop word list towards performance of information retrieval (IR). R. Puri et al., 2013[6] obtained a stop word list on Punjabi Language by searching most frequent present in the news articles from various popular Punjabi newspaper. Ashish et al., 2014 [7] used dictionary based stop word elimination technique in Gujarati language. Sharma & Jain, 2015[1] had shown the impact of stop words towards performances of text classification. Raulji et al., 2016 [8] used dictionary based approach where predefined list of stop words is compared to the target text on which removal is required which removes 13% words. Jha et al., 2016[9] used stop word removal algorithm for Hindi Language using the concept of a Deterministic Finite Automata (DFA). Siddiqi & Sharan, 2017[10] created stop words list for Hindi language with the help of linguistic experts. Rakholia & Saini 2017 [11] presented a rule-based approach to dynamically identify stop words for Gujarati language.

III. ASSUMPTIONS AND NOTATIONS

1. Let V = {t_1 , t_2 , ... , t_m} be a set of m terms or content words present in the corpus. Then documents can be represented as m-vectors $d_i = (w_{i1} , w_{i2} , ... , w_{im})$, where w_{ij} is the weight of term t_j in the document d_i , and the document collection or the corpus can be represented by a Term-Document Matrix D, where columns of D are document vectors $d_1 , d_2 , ... , d_n$ and rows of D are indexed by terms $t_1 , t_2 , ... , t_m$. Assuming each document d_i associated to one and only one class or category $c_i \in \{c_1 , c_2 , ... , c_k \}$, k<n e.g. Sports, Tourism, Literature etc. So by Pigeon Hole Principle more than one document belong to a single class.

We have used two weighing scheme :

Case1:

w_{ij} = tf-idf(t_j , d_i) = tf(t_j , d_i) * idf(t_j) …. eq. 1

Where, tf(t_j , d_i) = number of occurrences of the term t_j in a document d_i. where i=1, 2, … n and j=1, 2, … , m

idf(t_j) = log(n/df(t_j)) and df(t_j) = number of documents containing the term t_j …. eq. 2

Case2:

w_{ij} = tf-idf-icf(t_j , d_i) = tf-idf(t_j , d_i) *icf(t_j)[12] where i=1,2…n & j=1,2,…,m … eq. 3

icf(t_j) =log(k/cf(t_j)) and cf(t_j) = number of class containing the term t_j [12] …. eq. 4

2. Let us define a function, T(D) = A Set of all the terms present in a Term-Document Matrix D.

Therefore, |T(D)| = Number of unique term present in a Term-Document Matrix D.

IV. METHODOLOGY

Our POS(Parts-of-Speech) Tagged Bengali Corpus, Technology Development for Indian Languages Programme (TDIL), MeitY - Govt. of India, which is originally captured from various Bengali news papers containing news related to Sports, Tourism, Politics and Public Administration, Literature, Arts and Culture, Entertainment, Economy and Agriculture. It contains 49 documents and 127605 unique words or terms.

The algorithm for removal of semantically insignificant content words is divided into below steps:

Step 1. Removal of Special Characters : There are various special charters were present in the corpus which are treated as noise for text processing but present in Bengali literature like ▯ , - " ? ( ) ` ' : ! ; . These characters are first removed.

Step 2. Removal of Function Words : In a sentence, Pronouns, Prepositions, Conjunctions, Determiners, Qualifiers/Intensifiers, Auxiliary Verbs, and Interrogatives are defined as function words. On the other hand Nouns, Verbs, Adjectives, and Adverbs are defined as content words. Since we have POS(Parts-of-Speech) Tagged Corpus, we have selected words having parts of speech as one of Nouns, Verbs, Adjectives, and Adverbs and rest all are removed as functional words.

Step 3. Creation of Term-Document Matrix: At this stage, the content words present in each document of the given corpus. We have created Term-Document Matrix D for both Case1 and Case2 of Section III.

Step 4. Detecting Insignificant Content Words: The Term-Document Matrix D can be viewed as a collection of column vectors $\mathbf{t_1 , t_2 , ... , t_m}$ indexed by documents $d_1 , d_2 , ... , d_n$ where $\mathbf{t_j} = (w_{1j} , w_{2j} , ... , w_{mj})^T$.

If $\mathbf{t_j = 0}$ **,zero vector,** then t_j is detected as insignificant which is the basis for the algorithm.

V. RESULTS AND DISCUSSION

We prepared three data sets as described below. Each data set is modified version of the given Bengali corpus which is represented by a Term-Document Matrix along with class label for documents, as described in Section III. On each data set, we have executed different supervised Machine Learning(ML) like Decision Tree(J48), PART, NaiveBayesMultinomial, NaiveBayes, Optimized SVM (SMO). These Machine Learning(ML) algorithms are used for the text document classification. Each data set is split into training and testing data sets for those supervised Machine Learning(ML) algorithms using 10-split cross validation. The accuracy of each algorithm is captured and compared among the three data. The data sets are described below:

Data Set1 : The given corpus contains 127605 unique words. Step1 and Step2 of our algorithm removes 53371 special characters and functional words. Remaining 74234 words present in 49 text documents of the corpus represented as Data Set1 denoted by D_1. Data Set1 is the input to Step3 of our algorithm.

Data Set2 : If Step 3 of the algorithm creates a Term-Document Matrix based on tf-idf measure as specified in the Case1 of Section III, the algorithm detects 33 content words as insignificant. We remove these 33 insignificant content words from Data Set1. Remaining 74201 words present in 49

text documents of the corpus represented as Data Set2 denoted by D_2. Following content words are removed:

□ □\V□_VM_VNF , □ □ □\V□_V□VM_VF , □ □\RB , □ □ □V□_VM_VF , □ □ □JJ□ , □ □\V□_VM_VF , □ □ □\PR□_PRF , □ □ □\JJ □ , □ □ □V□_VM_VNG , □ □\V□_VAUX , □ □ □N□_NN , □ □\DM_DMD , □ □ □JJ□, □ □ □V□_VM_VNG , □ □\V□_VM_VNG , □ □ □V□_VM_VINF , □ □\V□_VAUX , □ □ □V□_VM_VF , □ □\V□_VAUX , □ □\DM_DMR , □ □\N□_NN , □ □V_VAUX , □ □ □V_VM_VINF , □ □ □\V□_V□VM_VINF , □ □ □V□_VM_VNG , □ □ □V□_VM_VNG , □ □\V□_VM_VNF , □ □\V□_VM_VNG , □ □ □ □\JJ □, □ □ □V□_VM_VINF , □ □DM_DMD , □ □DM_DMD , □ □\N□_NN

If a term t_j present in a all the documents, then $df(t_j) = n$, so $idf(t_j) = 0$ by eq.2. Therefore, $w_{ij} = 0$, i=1, 2, … n by eq.1. So, t_j is detected as insignificant at step 4 of our algorithm and removed from the corpus.

Data Set3 : If Step 3 of the algorithm creates a Term-Document Matrix based on tf-idf-icf measure as specified in the Case2 of Section III, the algorithm detects 618 content words as insignificant. We remove these 618 insignificant content words from Data Set1. Remaining 73616 words present in 49 text documents of the corpus represented as Data Set3 denoted by D_3.

If a term t_j present in a all the class then $df(t_j) = n$, so $icf(t_j) = 0$ by eq.4. Therefore $w_{ij} = 0$, i=1, 2, … n by eq.3. So, t_j is detected as insignificant at step 4 of our algorithm and removed from the corpus.

Table 1. Comparitive Analysis

ML Algorithm	Accuracy		
	Data Set1 (D_1)	Data Set2 (D_2)	Data Set3 (D_3)
Decision Tree(J48)	55.10%	55.10%	55.10%
PART	44.90%	44.90%	44.90%
NaiveBayesMultinomial	87.76%	89.80%	95.92%
NaiveBayes	81.63%	81.63%	81.63%
Optimized SVM (SMO)	79.59%	77.55%	77.55%

From the result it is clear that the accuracy remains same for three algorithms Decision Tree(J48), PART , NaiveBayes even after 33 content words removal in Data Set2 (D_2) and 618 content words removal in Data Set3 (D_3) from Data Set1 (D_1) by the proposed algorithm. However, the accuracy increases for NaiveBayesMultinomial with the removal of content words by the proposed algorithm. The accuracy decreases for SVM with the proposed algorithm for content word removal.

Also by the definition of Data Sets provided above, 33 content words removed from D_1 to create D_2.

Therefore,

$T(D_2) \subset T(D_1)$ … eq.5

and, $|T(D_1) - T(D_2)| = 33$ … eq.6

By Eq.5 & Eq.6 if we mark all 33 elements present in the set of terms $T(D_1) - T(D_2)$ as insignificant and remove them for further processing there will be no significance change in accuracy in text classification.

Also, 618 content words removed from D_1 to create D_3.

Therefore, $T(D_3) \subset T(D_1)$ … eq.7

and $|T(D_1) - T(D_3)| = 618$ … eq.8

By Eq.7 & Eq.8 if we mark all 618 elements present in the set of terms $T(D_1) - T(D_3)$ as insignificant and remove them for further processing there will be no significance change in accuracy in text classification.

Again, If $tf\text{-}idf(t_j, d_i) = 0$, then $tf\text{-}idf\text{-}icf(t_j, d_i) = 0$ by eq.3. It implies that the 33 content words, removed from D_1 to create D_2 , are also present in 618 content words, removed from D_1 to create D_3.

Therefore, $T(D_3) \subset T(D_2)$ … eq.9

and $|T(D_2) - T(D_3)| = 585$ … eq.10

Eq.9 implies that tf-idf-icf measure identifies the insignificant terms which is superset of those identified by tf-idf measure. Eq.10 implies that tf-idf-icf measure performs better than tf-idf in terms of textual noise removal.

VI. CONCLUSION AND FUTURE SCOPE

The proposed algorithm will remove the function and insignificant content words from large corpus. It approximately reduces 42% semantically insignificant terms or stop words from the corpus. If the corpus contains meaningful words associated with any special characters those might be filtered out at Step1 of the proposed algorithm. In future we may try with more larger corpus in Bengali as well as in other languages. Also the function words removed

by the algorithm may be used to add them in the list of Bengali stop words.

ACKNOWLEDGMENT

We acknowledge to Technology Development for Indian Languages Programme (TDIL), MeitY - Govt. of India and Indian Statistical Institute, Kolkata for their Bengali dataset support.

REFERENCES

[1] Dharmendra Sharma, Suresh Jain, "Evaluation of Stemming and Stop Word Techniques on Text Classification Problem", International Journal of Scientific Research in Computer Science and Engineering, Vol-3(2), PP (1-4) Apr 2015, E-ISSN: 2320-7639.

[2] Ljiljana Dolamic and Jacques Savoy, "When Stopword Lists Makethe Difference,", Journal of the American Society for Information Science and Technology no. 1, pp. 200–203, 2009.

[3] 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.

[4] R. Al-Shalabi, G. Kanaan, J. M. Jaam, A. Hasnah and E. Hilat, "Stop-word removal algorithm for Arabic language," Proceedings. 2004 International Conference on Information and Communication Technologies: From Theory to Applications, 2004., Damascus, Syria, 2004, pp. 545

[5] 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

[6] 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.

[7] Ashish T, Kothari M and Pinkesh P, "Pre-Processing Phase of Text Summarization Based on Gujarati Language", International Journal of Innovative Research in Computer Science & Technology (IJIRCST) Vol-2,Iss-4, July 2014

[8] Jaideepsinh K. Raulji, Jatinderkumar R. Saini, "Stop-Word Removal Algorithm and its Implementation for Sanskrit Language", International Journal of Computer Applications (0975 – 8887), Volume 150 – No.2, September 2016

[9] V. Jha, N. Manjunath, P. D. Shenoy and K. R. Venugopal, "HSRA: Hindi stopword removal algorithm," 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Durgapur, 2016, pp. 1-5

[10] 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.

[11] Rakholia R. M. and Saini J. R., "A Rule-based Approach to Identify Stop Words for Gujarati Language", accepted for publication in Advances in Intelligent and Soft Computing (AISC) Series, ISSN: 1615-3871, 2194-5357, 1860-0794 by Springer-Verlag, Germany. 2017.

[12] Ankita Dhar, Niladri Sekhar Dash, Kaushik Roy, "Categorization of Bangla Web Text DocumentsBased on TF-IDF-ICF Text Analysis Scheme", Springer Nature Singapore Pte Ltd. 2018,J. K. Mandal and D. Sinha (Eds.): CSI 2017, CCIS 836, pp. 477–484, 2018.

Authors Profile

Mr. Abhijit Barman pursed Bachelor of Engineering from Bengal Engineering and Science University, Shibpur, Howrah in 2004 and Master of Technology from Jadavpur University, Kolkata in year 2016. He is currently working as Associate Consultant at Tata Consultancy Services(TCS) Innovation Labs, Kolkata. He has around 14 years of Software(IT) working experience in Data Mining, Machine Learning, Pattern Recognition, Image Processing, Artificial Intelligence (AI) Systems Implementation, Real-time and Batch Analytics, Technology Consulting.

Dr. Diganta Saha is an eminent Professor of Department of Computer Science and Engineering at Jadavpur University, Kolkata. His field of specializations are Machine Translation, Language Engineering, Mobile Database Management, Text Processing, Text Classification and Text Data Mining. He has 2 Book chapters, 15 Journals and 54 Conferences publications. He is carried out 7 Projects under various funding agencies. Home Page: http://www.jaduniv.edu.in/profile.php?uid=660

