International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-2, December 2019

The Unprejudiced Stemmer to Prevent Etymological Behavior of Stemmed Morphemes Of Social Media Corpora

Akula .V.S. Siva Rama Rao, Ranjana .P

Abstract**: Sentiment Analysis is an application of Natural Langue Processing to analyze social media corpora to extract insights of corpora. Sentiment analytical results are the real feedback of the customers, which enables the organizations and companies to take appropriate decision on their products and business policies. Stemming plays in-evitable and vital role in sentiment analysis. Stemming is one of the phase of preprocessing the social media corpora. Today most of the researches uses strong stemmers to identify stem words of social media corpora. The most popular stemming algorithms such as Lancaster and Porter stemming algorithms causes prejudiced the meaning of the words. The over-stemmed words mislead the sentiment classification process. To prevent the over-stemming the Unprejudiced lighter stemming algorithm is proposed to sustain the meaning of the stemmed words. The propose Un-prejudiced algorithm uses lexical database and Parts of speech of Python Natural Language Tool Kit. There are a few stemming algorithm accuracy evaluation methods, in this paper we focused on Paice Error-rate relative to truncation (ERRT) measure to evaluate the accuracy of Lancaster, Porter and Unprejudiced stemming algorithms. The experiments were conducted on 25,758 source words and results were evaluated using Paice stem evaluation method and Sirsat method. The Paice Evaluation ERRT values 0.47209, 0.28703, 0.15502 of Lancaster, Porter, Unprejudiced respectively are proved that the Unprejudiced stemmer is more accurate than Lancaster and Porter. Sirsat's stem evaluation method Average Words Conflation Factor (AWCF) results 10310.31, 14031.17, 23349.87 of Lancaster, Porter, Unprejudiced respectively are also proved the Unprejudiced stemming algorithm is more accurate than Lancaster and Porter stemming algorithms.

***Keywords** : Sentiment Analysis, Social Media Corpora, Pre-processing, Etymology, Natural Language Processing, Stem Weight, Error-rate relative to truncation.*

I. INTRODUCTION

The huge social media network corpora emerged as major resource for Big Data Analytics.

Revised Manuscript Received on December 30, 2019.
***** Correspondence Author

Akula .V.S. Siva Rama Rao*, Research Scholar, Department of Computer Science and Engineering, Hindustan Institute of Technology and Science, Chennai, India. Associate Professor, Dept. of CSE, SITE, Tadepalligudem, AP, India

Ranjana. P, Professor, Department of Computer Science and Engineering, Hindustan Institute of Technology and Science, Chennai, India.

Sentiment Analysis analyze and quantify users textual views and opinions posted on the social media networks.

The social media datasets analytical results enable the organizations, companies and service centers to take vital decision accordingly[1][2][3][5] [16][17][22][23][30].Prior to apply the sentiment analysis algorithm, the social media corpora is under gone for text normalization process, where the tokens, which does not have any analytical value those tokens will be removed. Stemming is one of important phase of text normalization, where tweet words suffixes will be removed to identify the root word[14][20][28][29]. The stemming process may cause two types of errors one is under-stemming error and another is over-stemming error. The over-stemming error causes different meaning words conflate to the same word, so that ultimately its impact shows on sentiment analytical results[12][19][24].Stemming is used in various applications such as search performance tasks, Sentiment Analysis, Information Retrieval, reducing vocabulary space and Domain Analysis etc.[13][14][21][22][25]. Natural Language Processing(NLP) is subfield of Artificial Intelligence, which understand human languages and process through machine learning methods. Natural Languages Process can understand the sentiments of the users hidden under social media textual tweets and also classify them as positive and negative[1][2][4][5][6][13][18][19][20][26][27][28][30].*Stemming process is in-evitable in sentiment analysis, the strong Porter and Lancaster stemming algorithms reduce accuracy of the Sentiment Analysis as by the nature they removes more number of suffix letters from the stemming words and exhibits etymology behavior and causes over-stemming error[3][7][10][11][12][17][19][21][22][24][29].Etymological behavior Example.*

The different meaning words such as 'savings', 'savage' will be conflated to same word ie ' sav' in case Lancaster stemming. Similarly the Porter Stemming algorithm conflates the words 'patron' and 'patronize' into 'patron'[3][7][10][11][17][29].

To achieve balanced stemming the Unprejudiced stemming algorithm is proposed.

This paper divided into six sections, which are Introduction, Related works, Proposed Approach, Experiments and Evaluation, Conclusion & Future Work and References. In introduction section we discussed trends of sentiment analysis and flaws of stemming algorithms, objectives and proposed system.

In Related works state-of-the-art technologies were discussed and identified draw backs of over-stemming algorithms. We proposed Unprejudiced algorithm model in Proposed approach section. We adopt Paice-ERRT and Sirsat stemming accuracy evaluation methods to evaluate the results. In the Concluding section we emphasized ERRT accuracy metric. And finally listed out various reference papers in References section.

II. RELATED WORKS

Jose Luis Jimenez-Marquez Et al. (2018) in order to reduce the complexity to analyze social media corpora they developed two-stage framework. In the first "machine learning model" phase they setup the TFIDF Victimizer, where the data words were stemmed to their root form by using Python Snowball Stemmer from NLTK library to decrease the corpus size and to find important words in the text[1].Parama Fadli Kurnia and Suharjito (2018) created a business intelligence dashboard to analyze the performance of various topics and news posted on Facebook and Twitter social media. To aware the Topic of a news posts in face-book and twitter, they applied text classification techniques by using Naive Bayes, SVM and Decision Tree classification methods. In Content Analysis phase prior to applying the classification algorithm processing, the data preprocessing step have been taken up, in which they include data filtering, tokenizing, and stemming to avoid noise in the data sets[2].Rahardyan Bisma Setya Putra and Ema Utami(2018) stated that the Flexible affix classification stemmer unable to perform stemming on non-format affixed words of Indonesian languages, to perform stemming on these words they added new stemming rules to existing Nazief & Andriani stemming algorithm and obtained 73.3% of accuracy. They categorically stated that Porter Stemming algorithm is the standard stemming algorithm for English language[3]. Wael Etaiwi et.al(2018) discussed techniques used in Graph-based Arabic Natural Language processing and how the graph based techniques can use to resolve the NLP problems. Also discussed importance of text preprocessing normalization phases like tokenize, stop-words removal and Stemming to improve precision, recall and F-measure[4]. Chiraz LATIR .l (2018) developed social information system to deal verbose queries to extract very specific information. In which they applied morphosyntactic analysis to reduce verbose queries before submitting queries to the retrieval system. In the preprocessing phase they perform tasks like stop-words and stemming to reduce the verbose queries[6]. Andrei M. Butnaru and Radu Tudor Ionescu(2019) proposed an unsupervised and knowledge-based algorithm for Word Sense Disambiguation called ShotgunWSD2.0. Prior to apply ShotgunWSD2.0 they remove stopwords, applied Porter stemming algorithm on remaining words to eliminate most common morphological and in flexional endings[7].

Hiram Calvo, Arturo P. Rocha-Ramirez at.al(2019) Proposes a word senesce disambiguation model based on embedding representation of words using deep neural networks and obtained F1 Score 63.30.They used text processing tasks like convert text into lower case and applied Porter and Snowball stemming algorithms to remove suffixes[8].Axel Groß-Klußmann and at.al(2019) proposed Un-supervised and Supervised expert identification system to identify the major financial developments in economic regions and to predict profitable investments in stock market. They used Python NLTK to eliminate noise such as to removal of punctuations, stopwords, hashtags, casefolding, reduced the fraction of noise induced by informal language, applied Porter stemming algorithm on financial twitter datasets [10].

Vishal Vyas and V.Uma (2018) conducted experiments with Rapid Miner to analyze the tweets of sentiments and compared the accuracy levels with twenty different tools. They pre-processed the data in five steps: converting document to lower case, tokenization, filter stopwords, filter the word based on length and stemmed the words using Porter stemming algorithm[11].Jin Ding, Hailong Sun at.al(2018) developed an entity level sentiment analysis tool called "SentiSW", which contains sentiment classification and entity recognition which can classify the comments. They adopted preprocessing steps such as removing useless features and reduced the noise through words removal, words replacing and Snowball stemming[12].Mariem NEJI at.al(2018) proposed a semantic method to compose LingWs to give the support to the users to select a valid composition. Arabic language morphological level pre-process steps were included such as word segmentation, POS tagging, lemmatization and stemming[13].Prakruthi V , Sindhu D at.al(2018) evaluated the users sentiments about a person, product, brand and trend. They used twitter API to use tweets and built classification model and visualize result using histograms and pie charts. They taken up various preprocessing tasks, which includes tokenization, removal of unwanted words, special characters associated with usernames, hastags and stemming[14]. Doaa Mohey etal(2016) discussed various challenges of sentiment analysis and its evaluation during social media corpus analysis. They identified different obstacle to perform sentiment analysis on social media data, which includes Spam and fake, Domain dependence, Negation, World knowledge, NLP Overheads, Extracting features, Bi-polar and Huge lexicon. They emphasize need of improving accuracy of bi-polar ambiguous words and stemming in preprocessing phase[15].

III. THE PROPOSED APPROACH

A. Proposed Unprejudice Stemming Algorithm

The most popular Lancaster and Porter stemming algorithms are strong stemmers by nature they produce etymological behavior exhibits stemmed words[3][7][10][11][17][24][29]. To prevent etymology behavior of stemmed words we proposed light Unprejudiced(without damage) Stemming Algorithm.

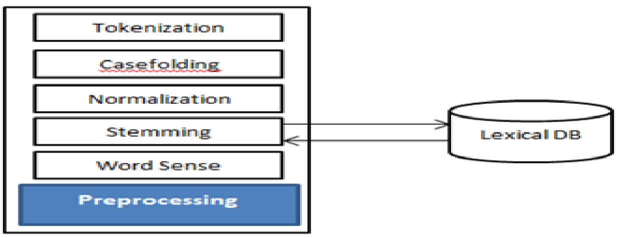

Fig 1 : Proposed Unprejudice Stemming Algorithm Architecture

The preprocessing involved various steps such as tokenization, casefolding, normalization, stemming and word sense as illustrated in the fig-1. Tokenization is splitting sentence into tokens, casefolding converting words into lowercase, normalization process is removing noise and unprejudiced stemming phase uses POS and lexical database to identify synonyms by using Python NLTK.[2][9][10][11][12][14][17][24][29][30].

B. Algorithm Implementation

To implement the unprejudiced algorithm we considered 25,758 source words consisting of all alphabetical words and organized as a groups as shown in the table-1 .

Table –1: Source words Grouping

Group Data Sets Before Stemming
thrones
throngs thronging thronged
throttles throttling throttled
throws throwing threw thrown
throw-ins
throwbacks
thrums thrumming thrummed
thrushes
thrusts thrusting
thuds thudding thudded
thugs
thumbs thumbing thumbed
thumbnails
thumbscrews
thumbtacks
thumps thumping thumped
thunders thundering thundered
thunderbolts
thunderclaps
thunderclouds
thunderstorms

To process the source words in the file all lines in the file were tokenized by using tokenize() function. Four different stems were generated, for each of the parts of speech by applying synonyms and lemmatize functions on the source word by using lexical corpus. Finally only one stem has been selected among four that have maximum number of synonyms and also checked whether final stem ends with 'ly', if so it was removed.

C.Unprejudice Stemming Algorithm

Un-Prejudice Stemming Algorithm (Python)

```
1.open(input_file) as file:
2.      read line ϵ file
// Split entire line into words(tokens)
3.           tokens ← tokenize the line
4.           read 'word' ϵ tokens
// Noun parts of speech synonym word
5.           num_of_nouns ← noun synonyms of the 'word'
6.           noun_stem ← find noun root of the 'word'
// Adverb parts of speech synonym word
7.           num_of_adverbs ← adverb synonyms of the 'word'
8.           adverb_stem ← find adverb root of the 'word'
// Verb parts of speech synonym word
9.           num_of_verbs ← verbs synonyms of the 'word'
10.          verb_stem ← find verb root of the 'word'
// Adjective parts of speech synonym word
11.          num_of_adjectives ← adject synonyms of the 'word'
12.          adjective_stem ← find adjective root of the 'word'
// Finding POS which have maximum frequency
13.          initialize stem ← noun_stem
14.                      max_num ← num_of_nouns
15.          if num_of_adverbs > max_num
16.                  max_num ← num_of_adverbs
17.                  stem ← adverb_stem
18.          if num_of_verbs > max_num
19.                  max_num ← num_of_verbs
20..                 stem ← verb_stem
21.          if num_of_adjectives > max_num
22.                  max_num ← num_of_adjectives
23.                  stem ← adjective_stem
// Remove if the stemmed words ends with 'ly'
24.            if stem.endswith('ly')
25.                  stem ← (replace 'ly' with null string)
// Read next word(token) from the line
26.       end_read_word
// Read next line from the file
27.     end_line_read
```

Table 2: Formatted stemmed Words

Formatted Groups After Unprejudice Stemming
'throne' : ['thrones']
'throng' : ['throngs', 'thronging']
'thronged' : ['thronged']
'throttle' : ['throttles', 'throttling', 'throttled']
'throw' : ['throws', 'throwing', 'threw', 'thrown']
'throw-in' : ['throw-ins']
'throwback' : ['throwbacks']
'thrum' : ['thrums', 'thrumming', 'thrummed']
'thrush' : ['thrushes']
'thrust' : ['thrusts', 'thrusting']
'thud' : ['thuds', 'thudding', 'thudded']
'thug' : ['thugs']
'thumb' : ['thumbs', 'thumbing', 'thumbed']
'thumbnail' : ['thumbnails']
'thumbscrew' : ['thumbscrews']
'thumbtack' : ['thumbtacks']
'thump' : ['thumps', 'thumping', 'thumped']
'thunder' : ['thunders', 'thundering', 'thundered']
'thunderbolt' : ['thunderbolts']
'thunderclap' : ['thunderclaps']
'thundercloud' : ['thunderclouds']
'thunderstorm' : ['thunderstorms']

The Table-2 shows the output of stemmed words after applying the unprejudiced algorithm the stemmed words with formatting that are to be used to evaluate using Paice formula[17][24].

D.Paice Stemming Strength Evaluation Model

The Paice proposed various evaluation metrics to assess stemming algorithms, the metrics include Under-Stemming index(UI), Over-Stemming index(OI) and Stem Weight(SW) and Error-rate relative to truncation(ERRT) [17][19][24].

Under-Stemming Index(UI) Under-Stemming Index will be calculated using the formula :

$$UI = \frac{GUMT}{GDMT}$$

Over-Stemming index(OI) Over-Stemming index will be calculated using the formula :

$$OI = \frac{GWMT}{GDNT}$$

Stemmer Weight : Stemmer Weight represents the strength of stemming algorithm, which is calculated with ratio of Over-stemming and Under-stemming. Stemmer Weight is calculated by using the formula :

$$SW = \frac{OI}{UI}$$

Error-rate relative to truncation (ERRT) : To find general relative accuracy of the stemming algorithms Paice proposed Error-rate relative to truncation (ERRT) measure. The ERRT can be computed using the following formula :

$$ERRT = \text{length (OP)/length (OT)}$$

E. Sirsat's Stemming Evaluation Method

Sirsat Et al. proposed stemming algorithms evaluation metrics to evaluate the stemming algorithms, such as Word Stemmed Factor, Correctly Stemmed Words Factor and Average Words Conflation Factor etc.[17][19][24].

Word Stemmed Factor (WSF) : Word Stemmed Factor is used to find strength of the stemmer. Word Stemmed Factor can be computed by using the formula :

$$WSF = \frac{WS}{TW} X100$$

*Correctly Stemmed Words Factor (CSWF)***:** The high *CSWF* value indicates higher accuracy of stemming algorithm. Correctly Stemmed Words Factor will be calculated using the formula :

$$CSWF = \frac{CSW}{WS} X100$$

Average Words Conflation Factor (AWCF) : The high value of AWCF represents high accuracy of stemming algorithm. The Average Words Conflation Factor will be calculated using the formula :

$$AWCF = \frac{CSW - NWC}{CSW} X100$$

IV. EXPERIMENTS AND EVALUATION

A.Paice's Stemming Evaluation

The Paice Stemming evaluation was used to evaluate the Unprejudice stemmer, where 25,758 samples stemming words were considered and divide into 14,760 groups. The 14,760 groups were divided into 15-datsets and performed the experiments using Lancaster, Porter and Unprejudice stemming algorithms and inferred results.

Table-3: 15-Datasets Under-stemming Index Average

Algorithm	UI-Average
Lancaster	0.142239267
Porter	0.139101133
Unprejudiced	0.149890333

Table-4: 15-Datasets Over-stemming Index Average

Algorithm	OI-Average
Lancaster	0.002377533
Porter	0.000615467
Unprejudiced	1.53333E-06

Table-5: 15-Datasets Stem-Weight Average

Algorithm	SW-Average
Lancaster	0.033240333
Porter	0.008643933
Unprejudiced	0.000011

Table-6: 15-Datasets Error-rate relative to truncation

Algorithm	ERRT-Average
Lancaster	0.472096733
Porter	0.2870392
Unprejudiced	0.1550206

The table Table-3 15-Datasets Under-stemming Index and Table-4 15-Datasets Over-stemming Index are used as parameters to compute Table-5 Stemmer Weight. Table-5 Stemmer Weight values consistently decrease from Lancaster to Unprejudiced stemmer, it is proved that the Unprejudiced stemmer is lighter than Lancaster and Porter stemming algorithms and also proved that Porter is lighter than Lancaster stemming algorithm.

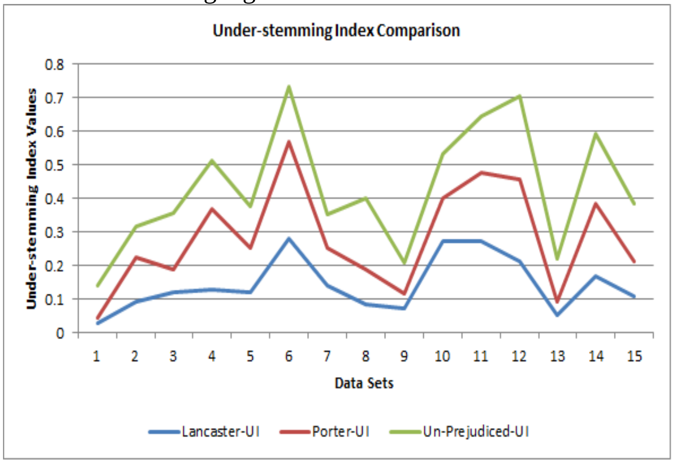

Fig-2 : Under-stemming Index Values Comparison

From the Fig-2 it is inferred that the Unprejudiced algorithm(High values) is more Under-stemming algorithm than both Lancaster and Porter, and also inferred that Porter is more Under-stemming algorithm than Lancaster algorithm.

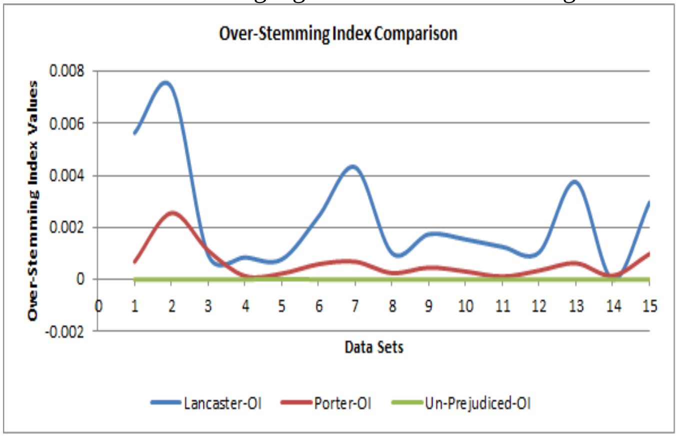

Fig-3 : Over-stemming Index Values Comparison

Fig-3 Inferred that the Lancaster algorithm(High values) is more over-stemmed algorithm than both Porter and Unprejudiced algorithms, and also inferred that Porter is more Over-stemmed algorithm than Unprejudiced algorithm.

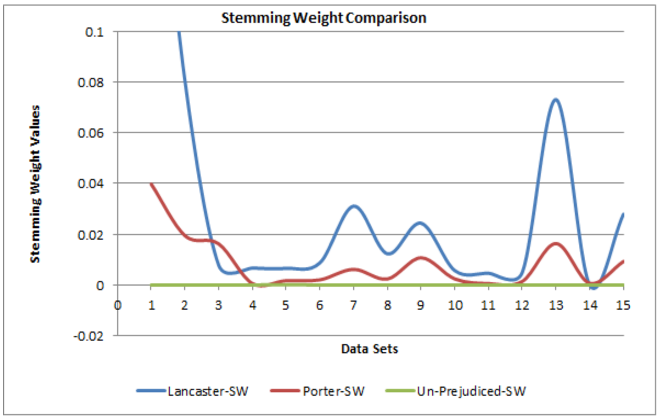

Fig-4 : Stemmer Weight Comparison

From the Fig-4, it is inferred that Unprejudiced algorithm(Low Values) is lighter than both Porter and Lancaster stemming algorithms, and also proved that Porter is lighter than Lancaster stemming algorithm.

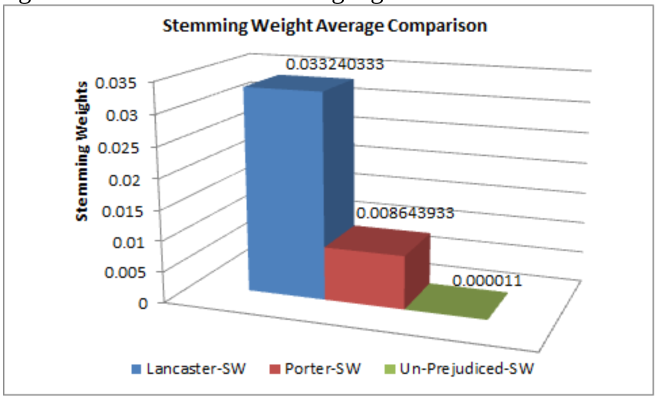

Fig-5 : 15-Datasets Stemmer Weight Average

From fig-5 15-datasets average stem weight differences proved that Unprejudiced algorithm is lighter(Low Values) than both Porter and Lancaster stemming algorithms and also proved that Porter is lighter than Lancaster stemming algorithm.

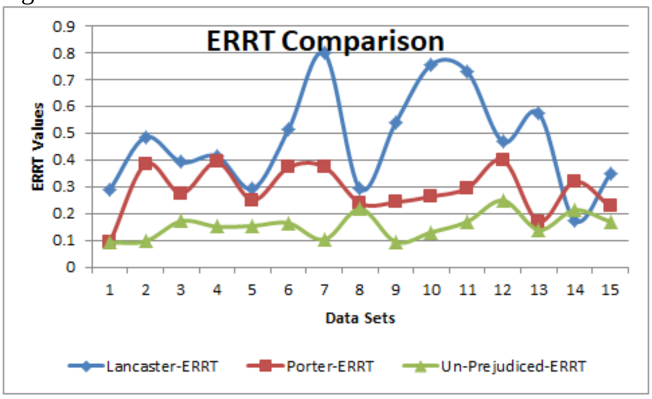

Fig-6 : Error-rate relative to truncationComparison

From the Fig-6, when the 15-datasets individual Error-rate relative to truncation(ERRT) results are compare with Lancaster, Porter and Unprejudiced algorithms, the Lancaster have highest ERRT values, the Porter stemmer algorithm have next higher ERRT values and finally the Unprejudiced algorithm have lowest ERRT. The low ERRT values represents more accurate than high ERRT. Therefore it is concluded that the Unprejudiced stemming algorithm is more accurate than other two algorithms.

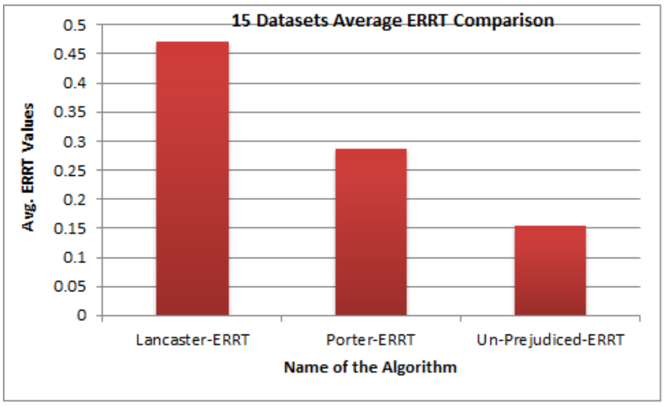

Fig-7: 15-Datasets Error-rate relative to truncation Comparison

With reference fig-7 the 15-datasets average of Error-rate relative to truncation also proved that the Unprejudiced stemming algorithm is more accurate than Lancaster and Porter stemming algorithms.

B.Sirsat's Stemming Evaluation

The Sirsat evaluation is another stemming evaluation method to assess the stemming algorithm accuracy. The same 25,758 words were used to evaluate Sirsat stemming evaluation and obtained Word Stemmed Factor (WSF), Correctly Stemmed Words Factor (CSWF) and Average Words Conflation Factor(AWCF) by using parameters Total Number of Words in the sample(TW), Number of Words Stemmed(WS), Correctly Stemmed Words (CSW), Number of distinct Stems after Stemming(S) and Number of Correct Words Not Stemmed(CW). The parameters and results were produced the table-7[17][24].

Table 7 : Sirsat's Stemming Evaluation

Evaluation Metric	Lancaster Stemming	Porter Stemming	Unprejudice Stemming
Total Number of Words sample(TW)	25758	25758	25758
Number of Words Stemmed(WS)	23638	23857	23788
Correctly Stemmed Words (CSW)	10414	14123	23409
Number of Distinct Stem after Stemming(S)	12918	14870	15812
Number of Correct Words Not Stemmed(CW)	2120	1901	1970
Word Stemmed Factor (WSF)	91.77	92.62	92.35
Correctly Stemmed Words Factor (CSWF)	44.06	59.20	98.41
Average Words Conflation Factor(AWCF)	10310.31	14031.17	23349.87

The Sirsat stemming evaluation Table-7 illustrates CSWF, AWCF values of Lancaster, Porter and Un-prejudice are monotonically increasing, which clearly states that the stemming accuracy increases along with their values.

V.CONCLUSION AND FUTURE WORK

Stemming algorithms are used to identify the root form of the word. Stemming is preprocessing phase of text normalization in Natural Language Processing, Language modeling and Information Retrieval System applications. The state-of-the-art of researches using Porter, Snowball and Lancaster algorithms for their applications. The implemented Unprejudiced stemming algorithm has proved that it is more accurate than Porter and Lancaster algorithms. There are a few evaluation methods to assess the stemming algorithms such as Paice and Sirsat methods. Our previous research stemming algorithm evaluation and other's evaluations limited to the stemming weight(SW), it can evaluate whether the stemmer is light-stemmer or strong-stemmer, but in this paper the Paice evaluation extended to Error-rate relative to truncation (ERRT), which evaluates the accuracy of stemming algorithm.The Paice's 15-datasets average ERRT values ie Lancaster : 0.472096733, Porter : 0.2870392 and Unprejudiced : 0.1550206, and Sirsat's Stemming evaluation CSWF resultant values Lancaster : 44.06, Porter : 59.20, Unprejudiced : 98.41 and AWCF Lancaster : 10310.31, Porter : 14031.17, Unprejudiced : 23349.87 resultants values proved that Un-prejudice Stemming algorithm is more accurate than both Lancaster and Porter Stemming algorithms.Un-prejudice Stemming algorithm can be applied where accuracy has higher priority than the retrieval. NLP-over heads and Domain dependency are other major obstacles of Sentiment Analysis, where research can be focused to improve the Sentiment Analytics.

REFERENCES

1. Jose Luis Jimenez-Marquez(2018), "Towards a big data framework for analyzing social media content", https://doi.org/10.1016/j.ijinfomgt.2018.09.003, 0268-4012/ © 2018 Elsevier Ltd. All rights reserved.
2. Parama Fadli Kurnia, Suharjito((2018), "Business Intelligence Model to Analyze Social Medi Information", 3rd International Conference on Computer Science and Computational Intelligence 2018, © 2018 The Authors. Published by Elsevier Ltd. ,https://creativecommons.org /licenses/by-nc- nd/4.0/
3. Rahardyan Bisma Setya Putra(2018),"Non-formal Affixed Word Stemming in Indonesian Language International Conference on Information and Communications Technology" ,978-1-5386-0954-5/18/ ©2018 IEEE
4. Wael Etaiwi and Arafat Awajan(2018), "Emirates Graph-based Arabic NLP Techniques:A Survey ", The 4th International Conference on Arabic Computational Linguistics Published by Elsevier B.V. http://creativecommons.org/licenses/by-nc-nd/3.0/
5. Sumithra, Velupiai,and Hanna Suominen(2018), "Using clinical Natural Language Processing for health outcomes research: Overview and actionable suggestions for future advances", 1532-0464/ © 2018 The Authors. Published by Elsevier Inc.
6. Mohamed ETTALEB, Chiraz LATIRI(2018) ," A Combination of Reduction and Expansion Approaches to Deal with Long Natural Language queries", 22nd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, Published by Elsevier.https://creativecommons.org/licenses/by-nc-nd/4.0/
7. Andrei, M., Butnaru and Radu Tudor(2019). ShotgunWSD 2.0: An Improved Algorithm for Global Word Sense Disambiguation. IEEE.,10.1109/ ACCESS .2019.2938058
8. Hiram Calvo, Arturo P. Rocha-Ramirez and Marco A. Morenoarmendariz and Carlos A. Duchanoy(2019). Toward Universal Word Sense Disambiguation Using Deep Neural Networks. IEEE., 10.1109/ACCESS.2019.2914921
9. Srishti Vashishtha and Seba Susan(2019). Fuzzy Rule based Unsupervised Sentiment Analysis from Social Media Posts. ELSEVIER., S0957-4174(19)30536-6
10. Axel Groß-Klußmann, Stephan König and Markus Ebner (2019). Buzzwords build momentum: Global financial Twitter sentiment and the aggregate stock market. Elsevier Expert Systems With Applications., 136 171–186.
11. Vishal Vyas and V.Uma(2018). An Extensive study of Sentiment Analysis tools and Binary Classification of tweets using Rapid Miner. Elsevier Procedia Computer Science., 125 329–335.
12. Jin Ding, Hailong Sun at.al(2018)." Entity-Level Sentiment Analysis of Issue Comments". ACM/IEEE 3rd International Workshop on Emotion Awareness in Software Engineering
13. Mariem NEJI, Bilel GARGOURI, Mohammed JMAIEL(2018). A semantic approach for constructing valid composition scenarios of linguistic Web services . Elsevier Ltd. https://creativecommons.org/licenses/by-nc-nd/4.0/
14. Prakruthi, Sindhu D Dr S Anupama Kumar(2018). Real Time Sentiment Analysis Of Twitter Posts. 3rd IEEE International Conference on Computational Systems and Information Technology for Sustainable Solutions 2018 ISBN: 978-1-5386-6078-2 © 2018 IEEE 29
15. Doaa Mohey El-Din Mohamed Hussein(2016),"A survey on sentiment analysis challenges" Elsevier B.V. on behalf of King Saud University. http://creativecommons.org/licenses/by-nc-nd/4.0.
16. Penubaka Balaji, D.Haritha(2018), "An Overview on Opinion Mining Techniques and Sentiment Analysis", International Journal of Pure and Applied Mathematics,Volume 118 No. 19 2018, 61-69 ISSN: 1311-ISSN: 1314-3395
17. Akula V.S. Siva Rama Rao, P. Ranjana,(2019). Empower Good Governance with Public Assessed Schemes by Improved Sentiment Analysis Accuracy. Inderscience DOI: 10.1504/EG.2020.10024136.
18. Wael Etaiwi and Ghazi Naymat(2017) , "The Impact of applying Different Preprocessing Steps on Review Spam Detection" , The 8th International Conference on Emerging Ubiquitous Systems and Pervasive Networks © 2017 The Authors. Published by Elsevier B.V
19. Felipe N. Flores, Viviane P. Moreira(2016), "Assessing the impact of Stemming Accuracy on Information Retrieval –A multilingual perspective", http://dx.doi.org/10.1016/j.ipm.2016.03.004 0306-4573/© 2016 Elsevier Ltd. All rights reserved.
20. Sercan Kulcu, Erdogan Dogdu(2016), "A Scalable Approach for Sentiment Analysis of Turkish Tweets and Linking Tweets To News", 978-1-5090-0662-5/16 $31.00 © 2016 IEEE
21. Mohamad Nizam Kassim, Mohd Aizaini Maarof(2016), "Word Stemming Challenges in Malay Texts: A Literature Review", Fourth International Conference on Information and Communication Technologies (ICoICT)ISBN: 978-1-4673-9879-4 (c) 2016 IEEE
22. Nourah F. Bin Hathlian Alaaeldin M. Hafezs(2016)," Sentiment - Subjective Analysis Framework for Arabic Social Media Posts", 978-1-4673-8956-3/16/ ©2016 IEEE
23. Demitrios E. Pournarakis(2016), "A Computational Model for Mining Consumer Perceptions in Social Media", A Computational Model for Mining Consumer Perceptions in Social Media, Decision Support Systems (2016), doi: 10.1016/j.dss.2016.09.018
24. Andrew Bimba, Norisma Idris, Norazlina Khamis ((2016), "Stemming Hausa text: using affix- stripping rules and reference look-up", Springer Science+Business Media Dordrecht, DOI 10.1007/s10579-015-9311-x
25. Tomaš Brychcin , Miloslav Konopik(2015), "HPS: High precision stemmer", Elsevier Ltd., Information Processing and Management 51 (2015) 68–91
26. Hongyuan Gao, Erin J. Aiello Bowles(2015),"Using natural language processing to extract mammographic findings" 2015 Elsevier Inc, www.elsevier.com/locate/yjbin
27. Abeed Sarker, Graciela Gonzalez ,(2015), "Portable automatic text classification for adverse drug reaction detection via multi-corpus training", Published by Elsevier Inc,(http://creativecommons.org/licenses/by-nc-nd/3.0/).
28. Célia Boyer, Ljiljana Dolamic, Gilles Falquet(2015), "Portable automatic text classification for adverse in automated detection of health websites' HONcode conformity: An Evaluation",. Published by Elsevier B.V. Peer-review under responsibility of SciKA – Association
29. Pragya Tripathi, Santosh Kr Vishwakarma, Ajay Lala (2015), "Sentiment Analysis of English Tweets Using RapidMiner", International Conference on Computational Intelligence and Communication Networks, 978-1-5090-0076-0/15 $31.00 © 2015 IEEE

30. Akula V.S. Siva Rama Rao, P. Ranjana,(2018)," Machine Learning Based Solution for Homograph and Auto-antonym Ambiguities in Social Media Corpora of Sentiment Analysis",Jour of Adv Research in Dynamical & Control Systems, Vol. 10, No. 3, 2018 ISSN 1943-023X 263.

AUTHORS PROFILE

Akula .V.S. Siva Rama Rao M.Tech., M.Phil., (Ph.D).
Pursuing Ph.D Programme from Hindustan Institute of Technology and Science, Chennai, India and working as Associate Professor in the Department of CSE, SITE, Tadepalligudem-534101, India.
Research interest areas including Big Data Analytics, Information Retrieval Systems and Data Mining.

Dr. P.Ranjana. M.E., Ph.D
Professor, Department of Computer Science and engineering, Hindustan Institute of Technology and Science, Chennai, India.
Areas of expertise including Computer networks, Design and Analysis of algorithms, Data structures, and Software Engineering.