Comparative Analysis of Effect of Corpus-Based Stemmers in Sentiment Analysis

Fahd Saleh Alotaibi[1], Vishal Gupta[2], Jasmeet Singh[3]
[1]*Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia*
[2]*University Institute of Engineering & Technology, Panjab University, Chandigarh, India*
[3]*University Institute of Engineering & Technology, Panjab University, Chandigarh, India*
(E-mail: [1]fsalotaibi@kau.edu.sa, [2]vishal@pu.ac.in, [3]jasmeetsingh.gagneja@gmail.com)

***Abstract*—**In Information Retrieval and Natural Language Processing applications, stemming is a common pre-processing tool. It improves the performance of the system by reducing the morphological variants of the word to the stem. The unsupervised corpus-based methods are preferred nowadays over linguistic stemmers due to the feature of the language independence. In this article, we analysed the effect of different strong corpus-based stemmers in the pre-processing phase of sentiment analysis task using two standard datasets related to movies review and product reviews. The results shows that corpus-based stemming is quite promising and resulted in large improvements in the classification accuracies as compared to no stemming baseline.

Keywords—Stemming; Corpus-Based; Stemmer; Sentiment Analysis

I. INTRODUCTION

Stemming is the linguistic process of grouping the similar words together. Usually, the morphologically related words are conflated together under the presumption that the morphologically related words are also semantically related to each other. Stemming is frequently employed in Natural Language Processing (NLP) and Information Retrieval (IR) tasks. In IR tasks, stemming increases both precision and recall as it helps in retrieving the documents at higher ranks that do not exactly match the terms of the query (Salton 1991, Kraaij and Pohlman 1996). Generally, all the documents which contain the words with same stem as the terms of the query are retrieved as relevant documents. In NLP applications, stemming decrease the size of feature set or index size and hence also reduces the complexity of the statistical models.

Sentiment analysis is an NLP application, which finds the perspective of the user from the given piece of text regarding a particular topic. Sentiment analysis is a classification task in which the text are classified to a particular pre-defined category (say positive or negative) according to the orientation of the opinion in the text. In classification task, stemming helps to reduce the complexity of the involved models. The statistical models are far less complex than what would have been if the original set of words were used. It also helps in better generalization as a small training error would result in a small test error (Bhamidipati and Pal 2007). Stemming in text mining can be considered as a feature selection or reduction mechanism where the major challenge is to choose the most appropriate dimension or features or concepts based on some grouping.

A number of stemming techniques have been proposed in the literature. The performance of the stemmers is the amount of reduction in the vocabulary size of the corpus. Aggressive stemmers reduce the vocabulary size for a given corpus drastically. The current stemming algorithms can be developed using language based stemming rules or using statistical or probabilistic methods. The language specific methods require prior knowledge about the morphology of the language as it makes use of stemming rules developed manually by the linguists or native speakers of the language. For highly agglutinative and morphologically complex languages, formation of these stemming rules is time consuming and quite tedious. The statistical methods of stemming, on the other hand, discovers morphologically related words from the corpus of language without any linguistic knowledge. Due to the nature of language independence, the statistical or corpus-based methods are now preferred choice of stemming.

The general objective of designing stemmers is to ensure improvement in retrieval performance, classification accuracy or performance of other NLP applications by handling the issue of vocabulary mismatch or complexity of statistical models. In this article, we present a comparative analysis of effect of different corpus-based stemmers on classification accuracies of sentiment or opinion mining datasets.

The rest of the article is organized as follows. In section 2, various corpus based stemmers described in the literature have been described. Section 3 describes the experiment setup and results of use of different corpus-based stemmers in classification of texts related to sentiments or opinion related documents. In Section 4, the analyses of results have been presented. Section 5 concludes this article.

II. CORPUS-BASED STEMMERS

A wide range of corpus-based methods has been proposed in the literature, These methods discover groups of morphologically related words from the corpus using unsupervised or semi-supervised techniques of learning. The first category of methods find the stem of each input word on the basis of the substring frequency. The first stemmer in this category has been described in Hafer and Weiss (1974). The proposed stemmer is based on the presumption that at the optimal break point of the stem and the ending, there is sharp

increase in the frequency of letter successor varieties. A variation of this technique has also been described in Stein and Pothast (2007). The concept of substring frequency is combined with Minimum Description Length principle to develop a corpus-based morphological analyser. Creutz and Lagus (2007) also employed MDL principle to develop a statistical model called Morfessor. Melucci et al. (2003) and Bacchin et al.(2005) also used frequency of substrings to create probabilistic models of stemmer generation.

The second category of corpus-based stemming techniques remove suffixes from the input word on the basis of the suffixes learnt from corpus using frequency based techniques. Oard (2001) developed suffix stripping method that removes suffixes up to length four characters. Paik and Parui (2011) also developed a similar stemmer that uses potential suffix knowledge along with common prefix information. Paik et al. (2011a) used graph-based clustering technique to group words on the basis of suffix knowledge.

The third category of corpus-based stemmers group words on the basis of string distance or similarity. Majumder et al. (2007) developed four similarity measures suitable for stemming and used them to group variant word forms. Ferna´ndez and Gutie´rrezrnandez (2011) used well-known edit distance to group variant words. Recently, Singh and Gupta (2017) developed a corpus-based method that conflates variants on the basis of Jaro-Winkler metric. Kasthuri et al. (2017) used string similarity feature along with dynamic programming technique to develop corpus-based stemmer. Chauvula and Suleman (2017) used weighted similarity metric for the purpose of corpus-based stemming.

Another category of corpus-based stemmers make use of co-occurrence or contextual knowledge to group morphological variant words. Xu and Croft (1998) and Paik et al. (2011b) developed method that uses co-occurrence strength to group variant words. Bhamidipati and Pal (2007) developed method that uses advanced form of co-occurrence called distributional similarity to group words. Peng et al. (2007) and Paik et al. (2013) used contextual knowledge to develop corpus-based stemmers suitable for the tasks of web searching or Information Retrieval. Brychcin and Konopik (2015) used semantic and lexical information for the purpose of stemming in various IR and NLP tasks.

III. CORPUS-BASED STEMMERS AND SENTIMENT ANALYSIS

Sentiment analysis is the process of automatically or computationally classifying the opinions, emotions or sentiments described in the piece of the text to analyse the attitude of the writer's regarding the product or the topic. There are a number of studies on classification where stemming has been used. These studies have varied results as some of the studies suggest stemming to be less beneficial in classification and some report that stemming is highly beneficial as it improves classification accuracy and reduce the dimensionality of the feature set. The studies by Riloff et al. (1995), Gustad et al. (2001), and Cohen et al. (2005) reported that stemming are quite advantages for the task of classification. In this section, we demonstrate the effect of corpus-based stemmers on the task of classification of texts related to sentiments or opinion mining. The following sub sections describe the experimental systems, evaluation metrics, data sets, baseline stemmers, and evaluation results.

A. Data Sets

The performance of different corpus-based stemmers for the task of sentiment analysis has been evaluated on the following data sets:

(1) Movie Dataset: This dataset contains 25,000 movie reviews, the dataset is available in both text and bag of words format. The dataset is available at http://ai.stanford.edu/~amaas/data/sentiment/.

(2) Amazon Product Review Dataset: This dataset contains reviews regarding different products. It contains data regarding product name, review texts, ratings etc. The dataset is available at https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html.

B. Experimental System and Evaluation Metrics

The classification experiments have been performed using the RTextTools toolkit provided in the R software. We used three different classifiers for the classification of texts namely SVM, Naïve Bayesian, and Maximum Entropy. The datasets are pre-processed using the different baseline corpus-based stemmers described in Section 3.3. The training and testing data for each dataset has been randomly divided and the same division has been used for all the corpus-based stemmers under analysis. In all the datasets, 60% data set has been used for training and 40% has been used for testing. The performance of classification is measured in terms of classification accuracy i.e. the number of correctly classified documents. Higher is the value of this parameter, better is the performance of the stemmer.

C. Baseline Corpus-Based Stemming Stratergies

In our analysis, we used seven different corpus-based stemmers in the experiments. The choice of the stemmers is such that they belong to different set of categories. Firstly, we used two stemmers namely LINGUISTICA and MORFESSOR which stems words on the basis of frequency of substrings and MDL principle. Secondly, we used two stemmers namely YASS (Yet Another Suffix Stemmer) and LEXSTEM (Lexical Stemmer) that uses string similarity metric to group morphologically related word forms. Thirdly, we used a stemmer named GRAS (GRAph based Stemmer) that groups words on the basis of suffix knowledge. Further, we used a co-occurrence based stemmer named SNS that makes use of a nearest neighbour clustering to group words. Lastly, we used a stemmer named HPS (High Precision Stemmer) that uses both lexical and co-occurrence knowledge between the words. Table I summarizes the baseline stemming strategies used in our experiments in terms of the abbreviation used, reference, source of code of stemmer (if any), and setting of parameters. The stemmers whose codes are not available has been implemented (reproduced) as per the method proposed by the authors of the stemmers.

TABLE I. SUMMARY OF BASELINE STEMMERS USED IN ANALYSIS

Abbreviation	Reference	Description	Parameter
LINGUISTICA[1]	GoldSmith (2001)	MDL and substring frequency based	Number of Tokens= 15,000,000
MORFESSOR[2]	Creutz and Lagus (2002)	MDL based	NIL
YASS[3]	Majumder et al. (2007)	String Similarity based	Clustering Cutoff = 1.5
GRAS	Paik et al. (2011a)	Suffix Knowledge based	Clustering Cutoff = 0.8
SNS	Paik et al. (2011b)	Co-occurrence Knowledge based	NIL
HPS[4]	Brychcin and Konopik (2015)	Lexical and Semantic Knowledge Based	Lexical Distance = 0.6
LEXSTEM	Singh and Gupta (2017)	String Distance based	Clustering Cutoff = 0.1

D. Evaluation Results

In this subsection, we present the experimental results of use of different corpus-based stemmers at the pre-processing stage of sentiment analysis. Table II and Table III present the effect of all stemmers on performance of SVM, Naïve Bayes and Maximum Entropy Classifiers for the movies and amazon dataset respectively. It is clear from the tables that all the stemmers under analysis improved the classification accuracies of all the stemmers against the no stemming baseline.

TABLE II. CLASSIFICATION ACCURACIES OF CLASSIFIERS WITH DIFFERENT STEMMERS FOR MOVIES DATA

METHOD	**SVM**	**Naïve Bayes**	**Maximum Entropy**
NO STEMMING	78.5%	79.5%	78.1%
LINGUISTICA	79.2%	79.7%	78.5%
MORFESSOR	81.2%	83.2%	81.2%
YASS	80.5%	82.8%	82.5%
GRAS	86.5%	85.4%	84.9%
SNS	82.1%	81.1%	82.4%
HPS	85.4%	84.2%	83.7%
LEXSTEM	87.2%	86.5%	86.0%

TABLE III. CLASSIFICATION ACCURACIES OF CLASSIFIERS WITH DIFFERENT STEMMERS FOR AMAZON DATA

METHOD	**SVM**	**Naïve Bayes**	**Maximum Entropy**
NO STEMMING	73.5%	72.7%	73.1%
LINGUISTICA	74.2%	71.7%	73.5%
MORFESSOR	79.3%	78.2%	78.2%
YASS	79.5%	79.8%	78.5%
GRAS	81.9%	81.4%	81.8%
SNS	80.1%	79.1%	80.4%
HPS	80.4%	80.2%	79.7%
LEXSTEM	82.4%	82.6%	82.1%

[1] Available at http://linguistica.uchicago.edu/linguistica.html
[2] Available at http://www.cis.hut.fi/projects/morpho/
[3] Available at http://www.isical.ac.in/~clia/resources.html
[4] Available at http://liks.fav.zcu.cz/HPS/

IV. ANALYSIS OF RESULTS

In this section, we critically analyse the results presented in Section 3.4. It is clear from the results all the corpus based stemmers showed improvement in classification accuracies of all the classifiers under analysis as compared to no stemming baseline (except LINGUISTICA in case of Naïve Bayes classifier). It is also depicted in Fig. 1 and Fig. 2.

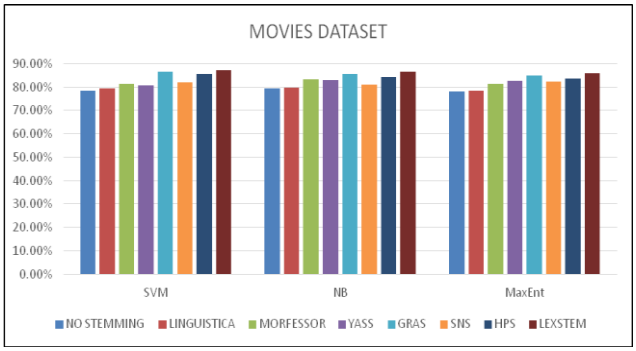

FIG. 1 COMPARISON OF DIFFERENT STEMMERS ON CLASSIFICATION ACCURACY FOR MOVIES DATA

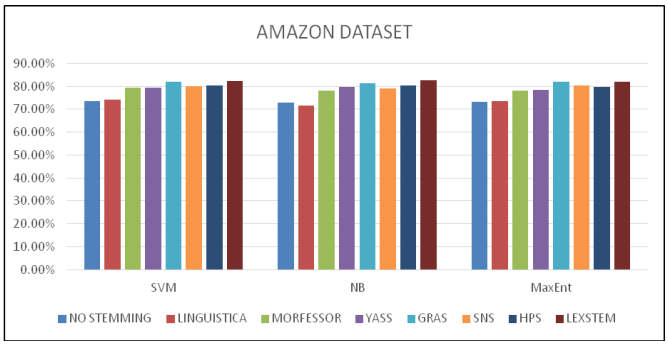

FIG. 2 COMPARISON OF DIFFERENT STEMMERS ON CLASSIFICATION ACCURACY FOR AMAZON DATA

Among all the corpus-based stemming techniques under analysis, LEXSTEM showed maximum improvement in classification accuracy among all the stemmers followed by GRAS. But GRAS tends to overstem the words. LINGUISTICA performed worse among all the stemmers and it take long time to develop classes of morphologically related words for small or moderate sized text collections also. MORFESSOR, YASS, and HPS performed almost equally. HPS increase precision at the small expense of recall. YASS falters when the suffixes are of long length.

V. CONCLUSION

In this article, we compared the performance of seven different state-of-the-art corpus based stemming techniques for the task of sentiment analysis. Two standard datasets namely movie review and Amazon product review has been used in the experiments. The result confirms that stemming is quite beneficial for the purpose of sentiment analysis. The results shows that corpus-based stemming is quite promising and resulted in large improvements in the classification accuracies as compared to no stemming baseline.

I. ACKNOWLEDGEMENT

This project was funded by Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, under grant no. (G: 201-611-1439). The authors therefore, acknowledge with thanks DSR technical and financial support.

REFERENCES

[1] Michela Bacchin, Nicola Ferro, and Massimo Melucci.2002. The Effectiveness of a Graph-Based Algorithm for Stemming. Springer-Verlag Berlin Heidelberg. pp 117–128.

[2] N. L. Bhamidipati , S. K. Pal, Stemming via Distribution-Based Word Segregation for Classification and Retrieval, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, v.37 n.2, pp 350-360, April 2007 [doi>10.1109/TSMCB.2006.885307]

[3] Tomáš Brychcín, Miloslav Konopík, HPS: High precision stemmer, Information Processing & Management, Volume 51, Issue 1, January 2015, pp 68-91

[4] Catherine Chavula and Hussein Suleman. 2017. Morphological Cluster Induction of Bantu Words Using a Weighted Similarity Measure. In Proceedings of the South African Institute of Computer Scientists and Information Technologists (SAICSIT '17). ACM, New York, NY, USA, Article 6, 9 pages. hŠ ps://doi.org/10.1145/3129416.3129453

[5] Mathias Creutz and Krista Lagus. 2002. Unsupervised Discovery of Morphemes. In Proceedings of the ACL-02Workshop onMorphological and Phonological Learning, pages 21–30.Association for Computational Linguistics.

[6] John Goldsmith, Unsupervised learning of the morphology of a natural language, Computational Linguistics, v.27 n.2, p.153-198, June 2001 [doi>10.1162/089120101750300490]M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[7] Goldsmith, J. 2005. An algorithm for the unsupervised learning of morphology. Tech. rep. TR-2005-06, Department of Computer Science, University of Chicago. http://humfs1.uchicago.edu/~jagoldsm/Papers/Algorithm.pdf.

[8] M. Kasthuri, S. B. R. Kumar and S. Khaddaj, "PLIS: Proposed Language Independent Stemmer for Information Retrieval Systems Using Dynamic Programming," *2017 World Congress on Computing and Communication Technologies (WCCCT)*, Tiruchirappalli, 2017, pp. 132-135. doi: 10.1109/WCCCT.2016.39

[9] Wessel Kraaij , Renée Pohlmann, Viewing stemming as recall enhancement, Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval, p.40-48, August 18-22, 1996, Zurich, Switzerland [doi>10.1145/243199.243209]

[10] Majumder, P., Mitra, M. and Datta, K. (2006), "Statistical vs. rule-based stemming for Monolingual French retrieval", working paper, CLEF 2006 Workshop, Alicante, 20-22 September, available at: http://clef.isti.cnr.it/2006/working_notes/workingnotes2006/majumderCLEF2006.pdf (accessed 24 September 2007). [Google Scholar]

[11] Melucci Massimo and Orio Nicola. "A novel method for stemmer generation based on hidden Markov models". Proceedings of the twelfth international conference on Information and knowledge management. 2003, 131-138.

[12] Oard DW, Levow G and Cabezas CI (2001) CLEF experiments at Maryland: Statistical stemming and back-off translation. In: Peters C, Ed. Proceedings of the First Cross-Language Evaluation Forum, pp. 176-187.

[13] Singh J, Gupta V. An efficient corpus-based stemmer. Cogn Comput. 2017;9(5):671–88.

[14] Jinxi Xu , W. Bruce Croft, Corpus-based stemming using cooccurrence of word variants, ACM Transactions on Information Systems (TOIS), v.16 n.1, p.61-81, Jan. 1998 [doi>10.1145/267954.267957]