Sparsity Adjusted Information Gain for Feature Selection in Sentiment Analysis

B.Y.Ong, S.W.Goh, Chi Xu
Planning and Operations Management Group
Singapore Institute of Manufacturing Technology (SIMTech), A*STAR
Singapore, Singapore
e-mail: {ongby, gary-goh, cxu}@simtech.a-star.edu.sg

Abstract—**The widespread use of social media and the internet are emerging trends that offer an additional interaction channel for companies to better understand customer sentiments about their brands and products. Sentiment analysis uses text data from social media such as customer comments and reviews, which has the nature of high dimensionality. Without selection, typically there are at least thousands of features (words or phrases) that can be extracted from a text corpus, among which there are many redundant or irrelevant features for sentiment classification task. Thus, it is critical to select a compact yet effective set of features to avoid the complex classifier design and slow running time of classification process. However, very few of existing metrics is able to improve efficacy of feature selection by addressing the issue of sparsity of feature matrix for text data, i.e., many features may appear only in a few documents. In this paper, an improved feature selection metric known as sparsity adjusted information gain (SAIG) is proposed, which modifies the conventional information gain metric and aims to adjust the feature ranking scores according to the sparsity of the feature vector. It is able to use less features to obtain a targeted performance level. The experiment results show that SAIG is able to improve the performance of sentiment classification.**

Keywords: Social media, sentiment analysis, feature selection, sparsity, information gain

I. INTRODUCTION

In this fast and ever changing digital era, the key to the future is to leverage on the new technologies to innovate creative and better ways to improve companies' current operating models. The rise of social media and the internet is one of the important aspects of the new information platform that provides opportunity for companies to tap on to better understand their customers. The usage of Big Data technologies offer the opportunities for improved data analytics to be conducted and to reveal more insights into the customers' behaviors and needs.

Social media has allowed individuals to interact with one another, build relationships, create brand awareness and improve customer service through social media services such as YouTube, Twitter, Yelp and Facebook [1]. Most importantly, social media allows consumers to easily share their opinions with just a click of a button. As a result, people are becoming more vocal by sharing their opinions online. On the receiving end, the demand for information for opinions about products has also been on a rise. It was concluded that 60% of Americans have referred to product reviews online for research at least once, with 15% of them doing so regularly on a daily basis, and about up to 87% of them recognizing the strong influence of the online reviews on their offline purchasing decisions [2, 3]. Major companies also come to realize the influence of online opinions have on their product sales. Although the companies do not have control over consumer-generated content, they can listen attentively to the consumer voices online so that they can respond quickly by adapting their marketing strategies, brand positioning and product development [4].

Sentiment analysis, sometimes also known as opinion mining, refers to the field of study which aims to automatically read and recognize human's "opinions, sentiments, evaluations, appraisals, attitudes and emotions towards entities such as products, services, organizations, individuals, issues, events, topics and their attributes" using computational means [5]. Sentiment analysis refers to the techniques to detect, or sometimes quantify the subjective viewpoint from a writer with respect to a certain subject. It is usually applied to large sets of unstructured data to gain an overview of the "public" opinions with respect to an entity of interest. Applications of sentiment analysis include (i) supporting demand prediction for products; (ii) evaluation of product performance to aspects level; (iii) monitoring of company brand and reputation; (iv) evaluation of marketing campaigns; and (v) customer market segmentation. The overall sentiment analysis process mainly includes three major steps: (1) data pre-processing, (2) feature selection, and finally, (3) classification.

Data pre-processing is the first and important step of any text mining analysis, where the data are cleaned and prepared for further analysis [6]. There are massive data that can be extracted from sources such as news, forum, blogs and social media. However, if these data are not properly cleaned, it can lead to misleading conclusion in the subsequent steps. The phrase "garbage in, garbage out" is very relevant in this case. In addition, data pre-processing is also the most time consuming phase of the

entire process, which might take up to almost 80% of the time of analysis [7, 8].

The next step is usually followed by feature selection, where it is the process of removing unnecessary features while keeping the features that have high differentiating abilities. As large data dimensionality is an inherent characteristic of text data, it is unavoidable that text data will have a huge set of features even after pre-processing. Hence, feature selection plays an important part in reducing the high dimensionality of the data, improving the efficiency performance of further analysis and minimizes computing memory requirements [9].

After dimension reduction, supervised classification uses the training dataset with labelled class to build the classifier. A testing data is transformed into a feature matrix defined by the selected features and fed into the classifier to assign a class label such as positive and negative. To verify the performance of the classification, some of the common performance metrics include accuracy, precision, recall and F1-score.

In this paper, the supervised classification approach is adopted. This approach has two sequential stages: training and testing. In training stage, a set of training data with known labels is required to build a training model for the supervised classifiers. In testing stage, supervised classifiers use training model to assign labels to each test data. The key idea is for the algorithms to detect and utilize on any pattern found in the training data and learn from the word frequency rates and/or syntactic structures, depending on the set of features selected. Subsequently, new online inputs are fed into the trained classification system. Figure 1 illustrates the entire flow of the sentiment analysis process.

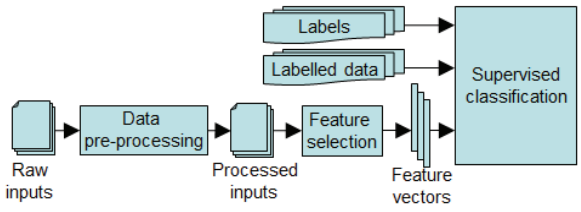

Figure 1. Flowchart of using sentiment analysis.

In this paper, an improved feature selection metric known as sparsity adjusted information gain (SAIG) is proposed to improve the quality of selected features, leading to an improved sentiment analysis classification performance. The rest of the paper is organized as follows: Section II covers the literature review of feature selections; Section III introduces the SAIG formulation and its algorithm; Section IV describes the dataset used and describes the detailed data pre-processing steps; Detailed results and discussions are given in Section V and finally Section VI concludes the paper and presents the future works.

II. LITERATURE REVIEW

Feature selection methods fall into two broad categories: wrapper methods and filter methods. Wrapper method requires one predetermined learning algorithm in feature selection and uses its performance to evaluate and select the subset of features. As for filter method, it relies on the training data to give a scoring metric to each of the features. After giving a scoring value to each of the features, the features are selected according to the ranking of the scoring values [10-12]. Some of the common feature selection metrics include document frequency (DF), term frequency (TF), term frequency-inverse document frequency (TF-IDF), information gain (IG), mutual information (MI) and χ^2 statistic (CHI). Some other metrics are accuracy (Acc), accuracy balanced (Acc2) and odds ratio (Odds) [9, 13-15]. As text data have high dimensionality, filter methods are more computational efficient than wrapper methods. The proposed method, SAIG falls into the category of filter methods.

For text data, features can be words or phrases. The number of possible words/phrases used in all test data can be very large and this number defines the length of the initial feature vector. However, for each text data such as a short comment in Twitter, the actual words/phrases used can be just a few, which means that there are many zeros in the feature vector if occurrence frequencies of words/phrases are used as values in the vector. Thus, the sparsity of the vectors can be very high.

A sparse matrix is one that contains a lot more zero values in the cells than non-zero values and is not very informative. Sparsity is the ratio of the number of zero cells to the total number of cells. If a matrix is 90% sparse, it is 10% dense. Fitting classifiers with a sparse matrix will degrade the performance of the text classification and adding more features does not guarantee better performance. The conventional feature selection metrics are unable to handle issue of sparsity properly.

The SAIG formulation introduced in this paper is adjusted according to the sparsity of the data and incorporates the term frequency. SAIG attempts to utilize all of the information (term frequency, document frequency and sparsity/density) that can be obtained from the sparse matrix to improve the feature selection. This allows reduction in the required total number of features to achieve a targeted performance. SAIG is modified from the conventional IG. Let $\mathcal{S}$ be the full set of features, $t_j \in \mathcal{S}$ be the j^{th} feature, $\mathbf{X}$ be the feature matrix, $x_{i,j}$ be the term frequency of t_j in the i^{th} document d_i, and c_a be class a. For binary classification, IG for feature t_j is given by [13]:

$$
\begin{aligned}
&IG(t_j) \\
&= -\sum_{a=1}^{2} P(c_a)\log(P(c_a)) \\
&+ P(t_j)\sum_{a=1}^{2} P(c_a|t_j)\log\left(P(c_a|t_j)\right) \\
&+ P(\bar{t}_j)\sum_{a=1}^{2} P(c_a|\bar{t}_j)\log\left(P(c_a|\bar{t}_j)\right) \qquad (1)
\end{aligned}
$$

where $P(c_a)$ is the probability of a document belonging to class c_a, $P(t_j)$ and $P(\bar{t}_j)$ are the probabilities of occurrence and non-occurrence of t_j in a document respectively, $P(c_a|t_j)$ is the conditional probability of a document belonging to class c_a given that t_j occurs in that document while $P(c_a|\bar{t}_j)$ is the counterpart of $P(c_a|t_j)$ and denotes the conditional probability of a document being class c_a given that t_j does not occur in it.

III. SPARSITY ADJUSTED INFORMATION GAIN

In this section, sparsity is formally defined and the deficiency of IG for feature selection is pointed out. Following the definition and discussion, SAIG metric is proposed.

Let N be the total number of documents associated with the feature matrix $\mathbf{X}$, N_{c_a} be the number of documents with class c_a, $DF_a(t_j)$ be the document frequency of the feature t_j in those documents belonging to class c_a. Denote $DF(t_j) = \sum_{a=1}^{2} DF_a(t_j)$ as the overall document frequency for t_j and let the feature vector for t_j be $\mathbf{x}_j = [x_{0,j}, x_{1,j}, \dots\dots, x_{N-1,j}]$. Based on the above definitions, the sparsity used throughout in this paper is defined as the probability $\beta(t_j) = P(t_j) = \sum_{a=1}^{2} \beta_a(t_j)$ where $\beta_a(t_j) = \frac{DF_a(t_j)}{N}$ for easier reference. High sparsity means $\beta(t_j) < \eta$ where η is a positive number much smaller than 1, i.e., $\beta(t_j) \ll 1$. High sparsity leads to only a few elements in $\mathbf{x}_j$ being non-zero.

The focus here is on a group of features which have spare vectors $\mathbf{x}_j$ and occur only in the documents of a particular class. This kind of features in c_a form a set $S_{r,c_a} = \{t_j : \beta(t_j) < \eta, t_j \in d_i \; with \; d_i \; belongs \; to \; c_a\}$. In the experiment (the details of experiments are given in Section IV), 66.64% and 78.83% of total features in two datasets are in such group which is a significant amount. These features may potentially have the discrimination power for classification but the conventional information gain metric cannot assign proper score to such feature. For instance, for those features with sparse vector and only occurring in class c_1, $P(c_1|t_j) = 1$ and $P(c_2|t_j) = 0$ are used in (1). These conditional probabilities are too strong for sparse features. A conditional probability value of 1 means that the document surely belongs to that class if this feature appeared. One of the root causes of this deficiency is that IG does not use term frequency which is more informative than document frequency.

The idea of SAIG is to heuristically adjust $P(c_a|t_j), a \in \{1,2\}$ in (1) based on term frequency and sparsity information. There are three cases for the adjustment. Case 1 and 2 are the cases when t_j only occurs in one class and its feature vector $\mathbf{x}_j$ is sparse. For SAIG, the conditional probabilities are adjusted for sparse features falling in only one class. For features that appear in high document frequencies, the strong probabilities are considered to be accepted. Case 3 handles the rest of the scenarios. Let $TF_a(t_j)$ be the term frequencies of the term t_j in class a and $TF(t_j) = \sum_{a=1}^{2} TF_a(t_j)$ be the term frequency in all documents. The details of three exclusive cases are as follows:

Case 1: $DF_1(t_j) > 0,\ DF_2(t_j) = 0, \beta(t_j) < \eta,$
Let $A = TF_1(t_j)$ & $B = \frac{0.5}{DF_1(t_j)}$
As $DF_1(t_j)$ increases, SAIG will converge to IG.

Case 2: $DF_1(t_j) = 0,\ DF_2(t_j) > 0, \beta(t_j) < \eta,$
Let $A = \frac{0.5}{DF_2(t)}$ & $B = TF_2(t_j)$
Case 2 is similar to case 1.

Case 3: For all other conditions,
Let $A = \frac{DF_1(t_j) * \frac{TF_1(t_j)}{TF(t_j)}}{DF(t_j)}$ & $B = \frac{DF_2(t_j) * \frac{TF_2(t_j)}{TF(t_j)}}{DF(t_j)}$

For the three cases, the two probabilities are calculated in the following way:

$$P(c_1|t_j) = \frac{A}{A+B}$$

$$P(c_2|t_j) = \frac{B}{A+B}$$

This ensures that the probabilities will be between 0 and 1. As the focus is on the feasibility of using term frequency, document frequency and sparsity to improve feature selection by modifying convention IG, an exemplary $\eta = 0.01$ is used for experiments. User can modify this value to see the relationship but in this analysis, the value of 0.01 is used as it is a common value or density threshold for sparse features. The SAIG algorithm is explained below.

SAIG Algorithm

Input: X, c_1, c_2, k, η

Output: The top k features and score metrics

for $j = 1$ *to* m **do**
  Check the case for t_j
  if $DF_1(t_j) > 0$ & $DF_2(t_j) = 0$ & $\beta(t_j) = \frac{DF_1(t_j)}{N} < \eta$
    $A = TF_1(t_j)$ & $B = \frac{0.5}{DF_1(t)}$
  else if $DF_1(t_j) = 0$ & $DF_2(t_j) > 0$ & $\beta(t_j) = \frac{DF_2(t_j)}{N} < \eta$
    $A = \frac{0.5}{DF_2(t)}$ & $B = TF_2(t_j)$
  else
    $A = \frac{DF_1(t_j) * \frac{TF_1(t_j)}{TF(t_j)}}{DF(t_j)}$ & $B = \frac{DF_2(t_j) * \frac{TF_2(t_j)}{TF(t_j)}}{DF(t_j)}$
  end
  Set $P(c_1|t_j) = \frac{A}{A+B}$ and $P(c_2|t_j) = \frac{B}{A+B}$
  Calculate the score for t_j
end
for $j = 1$ *to* m **do**
  Sort each feature t_j according to the score in descending order
  Select the top k features
end

IV. EXPERIMENT

The Amazon reviews dataset and another dataset that contains reviews of products, movies and restaurants are considered in this paper; the Amazon dataset is provided by Stanford Network of Analytics Platform, Stanford University [16]. The dataset consists of reviews from Amazon over a period of 18 years up to March 2013. For this paper, only a portion of product reviews from the "Beauty" product category are used. Since SAIG feature selection method is domain independent, it can be applied to other categories. Therefore, the use of one category is considered sufficient for this analysis. As for the other dataset, it is downloaded from UC Irvine Machine Learning Repository and created for the Paper 'From Group to Individual Labels using Deep Features' [17]. For easier reference, this dataset will be named as Movies dataset.

A. Datasets

The beauty product reviews file contains a total of 252,056 reviews, with each review containing the product identification code, the product title, product price, user identification code, user profile name, review helpfulness, review score (from a scale of 1-5), time of review, review summary and review text. As the text of the review is available in an unstructured form (a string of raw text), the format of the data is not suitable for further analysis, and needs to go through data pre-processing process to be transformed into a structured feature. The brief description of the particular pre-processing steps taken is outlined in the subsequent sub-section. Almost all of the reviews are in English, though there are a few reviews found to contain a mixed of English with other languages, but they are ignored as they appear in low frequencies to be deemed significant.

The Movies dataset contains sentences labelled with positive and negative sentiment. The dataset is well prepared and requires less pre-processing. Hence, the data pre-processing sequence will be referring to the Amazon dataset.

B. Data pre-processing sequence

From the raw data file, only the reviews text and the scores are extracted to be analyzed in this paper. The scores are used to be converted to a sentiment polarity based on a pre-selected threshold. In this paper, all reviews with a score of 3 and below are labelled as negative, and those with a score of 4 and above are labelled as positive.

As for the review text data, it is acknowledged that each individual researcher would apply a different sequence of data pre-processing steps. Therefore, the particular pre-processing sequence applied on the dataset will be briefing explained:

1) It was found that there are 49,678 duplicated reviews. Thus, in order to prevent these reviews from contaminating the true demographics of the reviews, they are removed from the dataset. The number of reviews in the dataset is 202,378 after this step.
2) It is observed that there is a presence of added repetitive letters or punctuations in some words, known as lengthening. These extra letters or punctuations might be added deliberately by the users to indicate additional emphasis of the emotion. In English, words containing sequences of three or more identical characters consecutively is not part of the standard dictionary and thus such words are highly probably to be the result of lengthening [18]. As the extent of lengthening varies and the differences of the lengths likely to be insignificant, a viable method to capture all lengthened words while minimizing sparseness and maintaining consistency is to replace all sequences of length three or greater to sequence of just three. For example, the word "yaaaaaayyyy!!!!!" is substituted by "yaaay!!!".
3) After step 2, the text is segmented into sentences so that analysis can be done with greater granularity. In this paper, a pre-trained Naïve Bayes sentence segmentation classifier is used using text from the corpus from the Penn Treebank project as the training dataset. The features used to detect the sentence boundaries are the capitalization of the next word and the presence of punctuations.
4) The text is further segmented into words, which will form the smallest granular meaningful feature

to be used in this analysis. A regular expression is defined to detect the boundaries of the words, which include whitespaces, special punctuations such as quotations and brackets.

5) A simple spell correction algorithm is applied to each word to correct misspells. The spelling correction algorithm is based on choosing the word with the minimum Damerau–Levenshtein edit distance using the Norvig's data as the training dataset [19].
6) During pre-processing, it is observed that there are a few occurrences of emoticons, also known as emotional icons, which is an emerging representation of expression via text on the internet. Emoticons are pictorial representation of facial expressions formed by typographical symbols such as ":)" and ":(" that communicates the mood of the author. In the pre-processing step, an emoticon detection regular expression is used [18], and once identified, they are substituted into their respective sentiment labels in words with reference from an emoticon lexicon [20].
7) Next, stop words removal which is one of the most standard text pre-processing steps is applied. Stop words are exceedingly common words such as "a", "the, "of" which do not contribute any semantic value to the process of sentiment analysis. These stop words are filtered out and discarded from the text, with the underlying assumption that these words do not carry any sentiment and have equal probability to appear in both positive and negative comments [21].
8) A common and critical piece of information to detect is negation. Negations are connotations that imply an opposite meaning with context to a statement. If negation is ignored, the recognized sentiment polarity of a classification may change drastically. In this paper, a technique by Das and Chen [22] is adopted by appending 'NOT_' to every word between a negation word (e.g. 'not', 'didn't) and the first punctuation (e.g. '.', '!', '?') following the negation word.
9) Lastly, in order to further reduce the sparsity of the data, stemming, another common text pre-processing step is applied. The stemming process transforms all inflected words into their derivative form, leaving behind only the shorter root form of a word. the commonly used Porter's English stemmer is chosen in the pre-processing step [23].

As the following classification step, 3 common classifiers are used. They are Naïve Bayes (NB), Support Vector Machine (SVM) and K-Nearest Neighbor (KNN).

V. Results and Discussion

As the focus is on the SAIG method, the Amazon dataset used in the analysis contains balanced classes of 618 documents and 3584 features with a sparsity index of 0.992. The Movies dataset contains 748 documents and 3142 features with a sparsity index of 0.997. The sparsity index is the ratio of zero cells to the total number of cells of the features matrix. TABLE I and TABLE II show the performance results for the Amazon dataset and the Movies dataset respectively. Each of the accuracy score is the average score after using 5-fold cross validation. Cross validation provides a better evaluation of the performance results. In general, the Movies dataset has a lower performance results due to the sarcasm contain in the movie reviews. Sarcasm is hard to detect and remains as a research topic. Looking at the performance results for the two datasets, SAIG performs better than IG when SVM and KNN classifiers are used. This is further validated with the higher F1 score for SAIG+SVM and SAIG+KNN when compared to IG with reference to Fig. 2 and Fig. 3. In addition, SAIG can use less features to obtain a targeted performance level with SVM and KNN. By comparing TABLE III and TABLE IV, SAIG selects better features in term of the absolute difference between the document frequencies of the two classes. Some features also suggest that the pre-processing step needs some improvements. A strong absolute difference means that the feature appears more often in one class than the other class and can be an important feature to build the classifier. Of course, there is case when the feature appears in one class and in low document frequency but yet an important feature. For the Amazon dataset, the word "Compliment" falls into the Case 1 scenario and is selected by SAIG as one of the top 10 features. As for IG, the word "Compliment" is ranked as the 50th feature. Another feature falling into the Case 2 scenario is the word "NOT_return". It is ranked as the 51st feature by SAIG but 142nd feature by IG. With proper pre-processing, the word "Compliment" will most likely represents a positive review. As for "NOT_return", it can mean that the customer will not return to the shop again and will be a good feature for negative review. As the data get larger, this consideration of Case 1 and Case 2 can assist in improving the feature selection. Comparing the 3 classifiers, it is noted that NB classifier has an unstable performance due to the strong independence assumptions between the features.

TABLE I. PERFORMANCE RESULTS FOR AMAZON DATASET

# of Features	Accuracy (%)					
	IG			*SAIG*		
	NB	***SVM***	***KNN***	***NB***	***SVM***	***KNN***
10	61.6	68.9	65.7	58.2	78.3	74.6
20	55.6	70.7	65.8	54.7	78.7	75.0
30	56.9	72.8	67.0	52.2	79.5	75.2
40	56.1	74.6	70.4	55.0	79.9	75.3
50	65.4	77.0	70.4	56.3	82.4	74.8
60	66.0	78.3	72.5	75.7	83.5	74.4
70	65.2	78.5	72.5	73.0	84.9	73.6
80	71.3	79.0	72.8	61.3	86.2	75.4
90	67.4	78.6	72.0	67.4	86.1	73.6
100	65.8	80.1	74.6	75.1	85.7	74.6

TABLE II. PERFORMANCE RESULTS FOR MOVIES DATASET

# of Features	Accuracy (%)					
	IG			*SAIG*		
	NB	***SVM***	***KNN***	***NB***	***SVM***	***KNN***
10	50.6	48.4	44.1	60.0	51.9	55.5
20	50.1	55.4	54.6	47.7	58.7	61.6
30	52.9	56.7	57.2	45.8	63.5	67.5
40	58.9	55.2	57.5	46.7	65.8	65.5
50	59.8	55.5	52.4	46.4	65.9	68.2
60	60.5	62.7	56.2	47.5	67.9	68.2
70	58.2	61.8	54.0	45.5	65.5	57.4
80	56.7	66.0	57.4	45.6	66.0	64.9
90	53.3	65.4	53.5	46.6	66.4	65.1
100	49.1	66.6	57.1	46.6	62.0	62.1

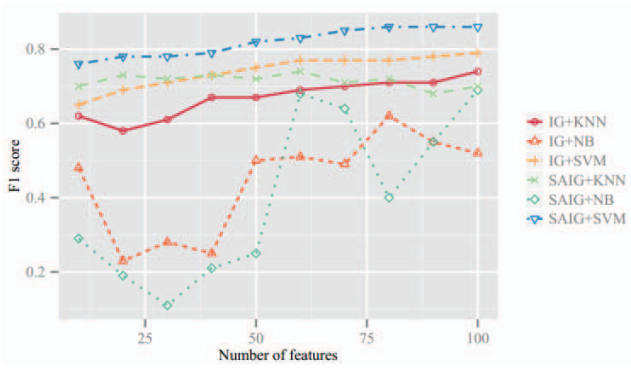

Figure 2. F1 score vs number of features for Amazon dataset.

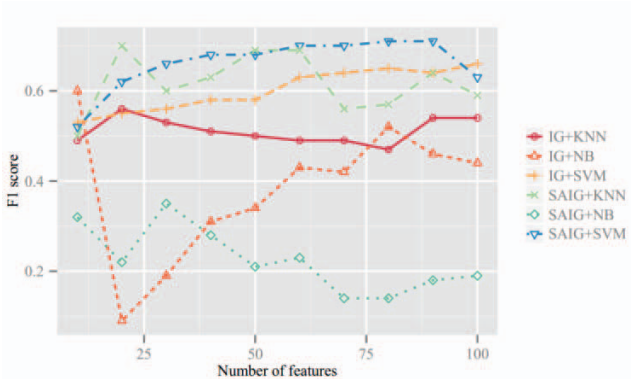

Figure 3. F1 score vs number of features for Movies dataset.

TABLE III. TOP 10 FEATURES SELECTED BY IG FOR AMAZON DATASET

Feature	$\mathbf{DF_1(t)}$	$\mathbf{DF_2(t)}$	$\mathbf{\lvert DF_1(t) - DF_2(t)\rvert}$
Smell	163	136	27
Perfum	135	116	19
Lik	107	90	17
Love	143	46	97
Scent	91	67	24
NOT_smell	65	75	10
Product	50	65	15
NOT_lik	51	63	12
Sweet	85	14	71
Bottl	39	60	21

TABLE IV. TOP 10 FEATURES SELECTED BY SAIG FOR AMAZON DATASET

Feature	$\boldsymbol{DF_1(t)}$	$\boldsymbol{DF_2(t)}$	$\boldsymbol{\lvert DF_1(t) - DF_2(t)\rvert}$
Love	143	46	97
Sweet	85	14	71
Sugar	78	13	65
Candi	73	7	66
Pink	69	11	58
Wear	59	20	39
Cotton	55	7	48
x.it	55	24	31
Compliment	48	0	48
Great	46	17	29

VI. CONCLUSIONS AND FURTURE WORK

Data sparsity is a common issue for textual data and will affect the performance of supervised classification. Data pre-processing and feature selection play important roles in working towards the improvement of the sentiment classification. It is viable to use any available information that exists in the data to improve the supervised classification. SAIG is able to perform better than IG with well-covered pre-processing steps for some classifiers by using information on term frequency, document frequency and sparsity. Data sparsity will remain an issue to the performance of the classifier. However, the next task is to research on how this performance results can be used to support demand prediction for products. This is the key added value that can assist the company. The performance results will not be perfect but the question is to what degree these results can be used to assist in demand prediction. A 75% performance accuracy may be relatively useful to support demand prediction. It is encouraged that companies share portion of their demand information to provide opportunities to explore the relationship between sentiment results and demand prediction.

ACKNOWLEDGMENT

This work is partially supported under the A*STAR TSRP fund 1424200021 and Antuit-SIMTech Supply Chain Analytics Lab.

REFERENCES

[1] L. Sheebarani, "Impact of Social Media on FMCG Advertising," *International Journal of Logistics & Supply Chain Management Perspectives,* vol. 2, pp. 541-546, 2014.

[2] J. Horrigan, "Online shopping," *Pew Internet & American Life Project Report,* vol. 36, 2008.

[3] (2007). *Online Consumer-Generated Reviews Have Significant Impact on Offline Purchase Behavior*. Available: http://www.comscore.com/Insights/Press-Releases/2007/11/Online-Consumer-Reviews-Impact-Offline-Purchasing-Behavior

[4] J. Zabin and A. Jefferies, "Social media monitoring and analysis: Generating consumer insights from online conversations," Aberdeen Group2008.

[5] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures on Human Language Technologies,* vol. 5, pp. 1-167, 2012.

[6] E. Haddi, *et al.*, "The Role of Text Pre-processing in Sentiment Analysis," *Procedia Computer Science,* vol. 17, pp. 26-32, 2013.

[7] K. Yiping, "A Survey on Preprocessing Techniques in Web Usage Mining " The Hong Kong University of Science and Technology, Hong Kong2003.

[8] D. Munková, *et al.*, "Data Pre-processing Evaluation for Text Mining: Transaction/Sequence Model," *Procedia Computer Science,* vol. 18, pp. 1198-1207, 2013.

[9] T. O'Keefe and I. Koprinska, "Feature selection and weighting methods in sentiment analysis," *ADCS 2009,* p. 67, 2009.

[10] M. A. Hall and L. A. Smith, "Feature Selection for Machine Learning: Comparing a Correlation-Based Filter Approach to the Wrapper," in *FLAIRS conference*, 1999, pp. 235-239.

[11] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research,* vol. 3, pp. 1157-1182, 2003.

[12] R. H. W. Pinheiro, *et al.*, "A global-ranking local feature selection method for text categorization," *Expert Systems with Applications,* vol. 39, pp. 12851-12857, 2012.

[13] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *ICML*, 1997, pp. 412-420.

[14] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *J. Mach. Learn. Res.,* vol. 3, pp. 1289-1305, 2003.

[15] Ş. Taşcı and T. Güngör, "Comparison of text feature selection policies and using an adaptive framework," *Expert Systems with Applications,* vol. 40, pp. 4871-4886, 2013.

[16] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text," in *Proceedings of the 7th ACM conference on Recommender systems*, 2013, pp. 165-172.

[17] D. Kotzias, *et al.*, "From Group to Individual Labels using Deep Features," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 597-606.

[18] C. Potts. (2011). *Sentiment Symposium Tutorial: Tokenizing*. Available: http://sentiment.christopherpotts.net/tokenizing.html

[19] P. Norvig. (n.d.). *How to Write a Spelling Corrector*. Available: http://norvig.com/spell-correct.html

[20] A. Agarwal, *et al.*, "Sentiment analysis of twitter data," in *Proceedings of the Workshop on Languages in Social Media*, 2011, pp. 30-38.

[21] S. Roy, *et al.*, "A Lexicon Based Algorithm For Noisy Text Normalization As Pre-processing For Sentiment Analysis," *International Journal of Research in Engineering and Technology,* vol. 2, pp. 67-70, 2013.

[22] S. Das and M. Chen, "Yahoo! for Amazon: Extracting market sentiment from stock message boards," in *Proceedings of the Asia Pacific finance association annual conference (APFA)*, 2001, p. 43.

[23] R. Boulton and M. Porter. (2006). *Snowball*. Available: http://snowball.tartarus.org/