On Stopwords, Filtering and Data Sparsity for Sentiment Analysis of Twitter

Hassan Saif,[1] **Miriam Fernandez,**[1] **Yulan He,**[2] **Harith Alani**[1]
[1]Knowledge Media Institute, The Open University, UK
{h.saif, m.fernandez, h.alani}@open.ac.uk
[2]School of Engineering and Applied Science, Aston University, UK
{y.he@cantab.net}

Abstract

Sentiment classification over Twitter is usually affected by the noisy nature (abbreviations, irregular forms) of tweets data. A popular procedure to reduce the noise of textual data is to remove stopwords by using pre-compiled stopword lists or more sophisticated methods for dynamic stopword identification. However, the effectiveness of removing stopwords in the context of Twitter sentiment classification has been debated in the last few years. In this paper we investigate whether removing stopwords helps or hampers the effectiveness of Twitter sentiment classification methods. To this end, we apply six different stopword identification methods to Twitter data from six different datasets and observe how removing stopwords affects two well-known supervised sentiment classification methods. We assess the impact of removing stopwords by observing fluctuations on the level of data sparsity, the size of the classifier's feature space and its classification performance. Our results show that using pre-compiled lists of stopwords negatively impacts the performance of Twitter sentiment classification approaches. On the other hand, the dynamic generation of stopword lists, by removing those infrequent terms appearing only once in the corpus, appears to be the optimal method to maintaining a high classification performance while reducing the data sparsity and substantially shrinking the feature space.

Keywords: Sentiment Analysis, Stopwords, Data Sparsity

1. Introduction

Sentiment analysis over Twitter has recently become a popular method for organisations and individuals to monitor the public's opinion towards their brands and business. One of the key challenges that Twitter sentiment analysis methods have to confront is the noisy nature of Twitter generated data. Twitter allows only for 140 characters in each post, which influences the use of abbreviations, irregular expressions and infrequent words. This phenomena increases the level of data sparsity, affecting the performance of Twitter sentiment classifiers (Saif et al., 2012a).

A well known method to reduce the noise of textual data is the removal of stopwords. This method is based on the idea that discarding non-discriminative words reduces the feature space of the classifiers and helps them to produce more accurate results (Silva and Ribeiro, 2003). This pre-processing method, widely used in the literature of document classification and retrieval, has been applied to Twitter in the context of sentiment analysis obtaining contradictory results. While some works support their removal (Bakliwal et al., 2012; Pak and Paroubek, 2010; Zhang et al., 2012; Speriosu et al., 2011; Gokulakrishnan et al., 2012; Kouloumpis et al., 2011; Asiaee T et al., 2012) others claim that stopwords indeed carry sentiment information and removing them harms the performance of Twitter sentiment classifiers (Saif et al., 2012b; Hu et al., 2013b; Martınez-Cámara et al., 2013; Hu et al., 2013a).

In addition, most of the works that have applied stopword removal for Twitter sentiment classification use pre-compiled stopwords lists, such as the *Van stoplist* (Rijsbergen, 1979), the *Brown stoplist* (Fox, 1992), etc. However, these stoplists have been criticised for: (i) being outdated (Lo et al., 2005; Sinka and Corne, 2003b) (a phenomena that may affect specially Twitter data, where new information and terms are continuously emerging) and, (ii) for not accounting for the specificities of the domain under analysis (Ayral and Yavuz, 2011; Yang, 1995), since non-discriminative words in some domain or corpus may have discriminative power in different domain.

Aiming to solve these limitations several approaches have emerged in the areas of document retrieval and classification that aim to dynamically build stopword lists from the corpus under analysis. These approaches measure the discriminative power of terms by using different methods including: the analysis of terms' frequencies (Trumbach and Payne, 2007; Lo et al., 2005), the term entropy measure (Sinka and Corne, 2003b; Sinka and Corne, 2003a), the Kullback-Leibler (KL) divergence measure (Lo et al., 2005), and the Maximum Likelihood Estimation (Ayral and Yavuz, 2011). While these techniques have been widely applied in the areas of text classification and retrieval, their impact in Twitter sentiment classification has not been deeply investigated.

In this paper we aim to study the effect of different stopword removal methods for polarity classification of tweets (positive vs. negative) and whether removing stopwords affects the performance of Twitter sentiment classifiers. To this end, we apply six different stopword removal methods to Twitter data from six different datasets (obtained from the literature of Twitter sentiment classification) and observe how removing stopwords affects two well-known supervised sentiment classification methods, Maximum Entropy (MaxEnt) and Naive Bayes (NB). We assess the impact of removing stopwords by observing fluctuations on: (i) the level of data sparsity, (ii) the size of the classifier's feature space and (iii), the classifier's performance in terms of accuracy and F-measure.

Our results show that pre-compiled stopword lists (classic stoplists) indeed hamper the performance of Twitter senti-

ment classifiers. Regarding the use of dynamic methods, stoplists generated by mutual information produce the highest increase in the classifier's performance compared to not removing stopwords (1.78% and 2.54% average increase in accuracy and F-measure respectively) but a moderate reduction on the feature space and with no impact on the data sparsity. On the other hand, removing singleton words (those words appearing only once in the corpus) maintain a high classification performance while shrinking the feature space by 65% and reducing the dataset sparsity by 0.37% on average. Our results also show that while the different stopword removal methods affect sentiment classifiers similarly, Naive Bayes classifiers are more sensitive to stopword removal than the Maximum Entropy ones.

The rest of the paper is organized as follows. Related work is presented in Section 2, and our analysis set up is presented in Section 3. Evaluation results are presented in Section 4. Discussion and future work are covered in Section 5. Finally, we conclude our work in Section 6.

2. Related Work

Stopwords, by definition, are meaningless words that have low discrimination power (Lo et al., 2005). The earliest work on stopwords removal is attributed to Hans Peter Luhan (1957), who suggested that words in natural language texts can be divided into keyword terms and non-keyword terms. He referred to the latter as *stopwords*. Inspired by Luhan's work, several pre-complied stoplists have been generated such as the *Van stoplist* (Rijsbergen, 1979), which consists of 250 stopwords and the *Brown stoplist* (Fox, 1992), which consists of 421 stopwords. These lists are usually known as the *classic* or the *standard* stoplists.

Despite their popularity, the classic stoplists face two major limitations: (i) they are outdated in the sense that they do not cover new emerging stopwords on the web (Lo et al., 2005; Sinka and Corne, 2003a; Sinka and Corne, 2003b), and (ii) they are too generic and provide off-topic and domain-independent stopwords. Also, their impact on the feature space tends to be limited since they often consists of a small number of words (Yang, 1995).

To overcome the above limitations, several methods to automatically generating stoplists have recently emerged. They can be categorised into: methods based on zipf's law (Trumbach and Payne, 2007; Makrehchi and Kamel, 2008; Forman, 2003) and methods based on the information gain criteria (Lo et al., 2005; Ayral and Yavuz, 2011).

Zipf (1949) observed that in a data collection the frequency of a given word is inversely proportional to its rank. Inspired by this observation (aka Zipf's Law), several popular stopword removal method have been explored in the literature. Some methods assume that stopwords correspond to those of top ranks (i.e., most frequent words) (Lo et al., 2005; Trumbach and Payne, 2007), while others consider both top- and low-ranked words as stopwords (Makrehchi and Kamel, 2008). The inverse document frequency (*IDF*) has also been explored in the literature as another popular variation to using the raw frequency of words (Forman, 2003; Lo et al., 2005).

The information gain ranking methods rely on the amount of information that terms have in text. The notion behind this is that stopwords are those who have very low informativeness values. Several measures have been used to calculate the informativeness power of words including: the term entropy measure (Sinka and Corne, 2003b; Sinka and Corne, 2003a), the Kullback-Leibler (KL) divergence measure (Lo et al., 2005), and the Maximum Likelihood Estimation (Ayral and Yavuz, 2011).

From the above review, one may notice that the problem of stopwords generation and removal has been extensively researched in various areas such as information retrieval, text classification and machine translation. As for Twitter sentiment analysis, however, the area still lacks a proper and deep analysis of the impact of stopwords on the sentiment classification of tweets. To fill up this gap we focus our research in this paper towards experimenting and evaluating several of-the-shelf stoplist generation methods on data from various Twitter corpora as will be described in the subsequent sections.

3. Stopword Analysis Set-Up

As mentioned in the previous sections, our aim is to assess how different stopword removal methods affect the performance of Twitter sentiment classifiers. To this end, we assess the influence of six different stopword removal methods using six different Twitter corpora and two different sentiment classifiers. The complete analysis set up is composed by:

3.1. Datasets

Stopwords may have different impact in different context. Words that do not provide any discriminative power in one context may carry some semantic information in another context. In this paper we study the effect of stopword removal in six different Twitter datasets obtained from the literature of Twitter sentiment classification:

- The Obama-McCain Debate dataset (OMD) (Shamma et al., 2009).
- The Health Care Reform data (HCR) (Speriosu et al., 2011).
- The STS-Gold dataset (Saif et al., 2013).
- Two datasets from the Dialogue Earth project (GAS, WAB) (Asiaee T et al., 2012).
- The SemEval dataset (Nakov et al., 2013).

Table 1 shows the total number of tweets and the vocabulary size (i.e., number of unique word unigrams) within each dataset. Note that we only consider the subsets of positive a negative tweets from these datasets since we perform binary sentiment classification (positive vs. negative) in our analysis.

3.2. Stopword removal methods

The *Baseline method* for this analysis is the non removal of stopwords. In the following subsections we introduce the stopword removal methods we use in our study.

Dataset	No. of Tweets	Vocabulary Size
OMD	1,081	3,040
HCR	1,354	5,631
STS-Gold	2,034	5,780
SemEval	5,387	16,501
WAB	5,495	10,920
GASP	6,285	12,828

Table 1: Statistics of the six datasets used in this paper

The Classic Method
This method is based on removing stopwords obtained from pre-compiled lists. Multiple lists exist in the literature (Rijsbergen, 1979; Fox, 1992). But for the purpose of this work we have selected the classic Van stoplist (Rijsbergen, 1979).

Methods based on Zipf's Law *(Z-Methods)*
In addition to the classic stoplist, we use three stopword generation methods inspired by Zipf's law including: removing most frequent words (*TF-High*) and removing words that occur once, i.e. singleton words (*TF1*). We also consider removing words with low inverse document frequency (*IDF*). To choose the number of words in the stoplists generated by the aforementioned methods, we first rank the terms in each dataset based on their frequencies (or the inverse document frequencies in the IDF method). Secondly, we plot the rank-frequency distribution of the ranked terms. The size of the stoplist corresponds to where an "elbow" appears in the plot. For example, Figure 1 shows the rank-frequency distribution of terms in the GASP dataset with the upper and lower cut-offs of the elbow in the distribution plot. From this Figure, the TF-High stoplist is supposed to contain all the terms above the upper cut-off (50 terms approximately). On the other hand, the TF1 stoplist should contain all the terms below the lower cut-off.

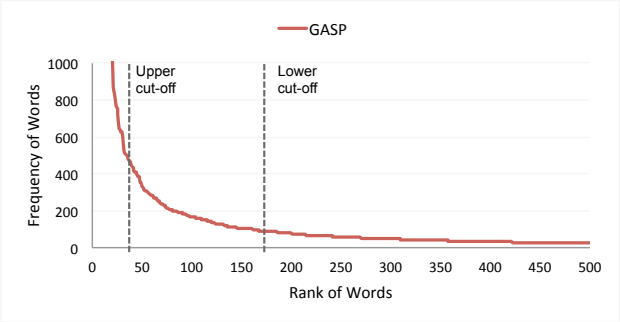


Figure 1: Rank-Frequency distribution of the top 500 terms in the GASP dataset. We removed all other terms from the plot to ease visualisation.

Figure 2 shows the rank-frequency distribution of terms for all datasets in a log-log scale. Although our datasets differ in the number of terms they contain, one can notice that the rank-frequency distribution in all the six datasets fits well the Zipf distribution.

Term Based Random Sampling (*TBRS*)
This method was first proposed by Lo et al. (2005) to automatically detect stopwords from web documents. The

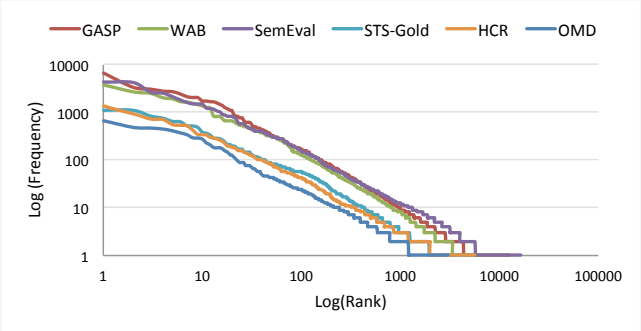


Figure 2: Frequency-Rank distribution of terms in all the datasets in a log-log scale

method works by iterating over separate chunks of data randomly selected. It then ranks terms in each chunk based on their informativeness values using the Kullback-Leibler divergence measure (Cover and Thomas, 2012) as shown in Equation 1.

$$d_x(t) = P_x(t).log_2 \frac{P_x(t)}{P(t)} \quad (1)$$

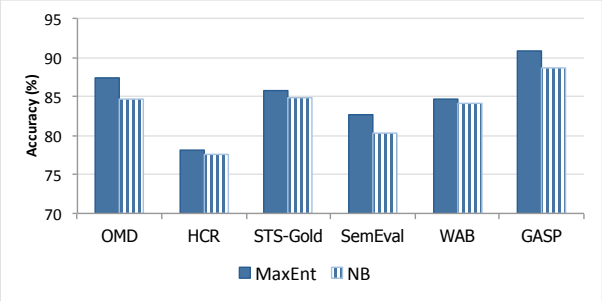
where $P_x(t)$ is the normalised term frequency of a term t within a chunk x, and $P(t)$ is the normalised term frequency of t in the whole collection.
The final stoplist is then constructed by taking the least informative terms in all chunks, removing all possible duplications.

The Mutual Information Method (*MI*)
Stopwords removal can be thought of as a feature selection routine, where features that do not contribute toward making correct classification decisions are considered stopwords and got removed from the feature space consequently. The mutual information method (MI) (Cover and Thomas, 2012) is a supervised method that works by computing the mutual information between a given term and a document class (e.g., positive, negative), providing an indication of how much information the term can tell about a given class. Low mutual information suggests that the term has low discrimination power and hence it should be easily removed.
Formally, the mutual information between two random variables representing a term t and a class c is calculated as (Xu et al., 2007):

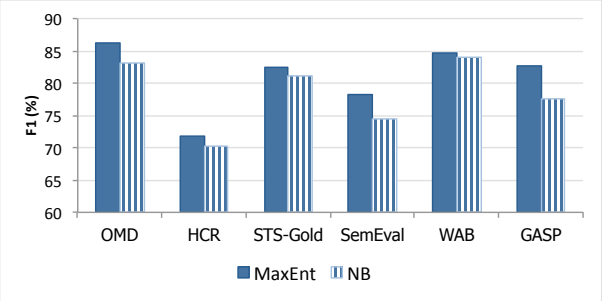
$$I(T;C) = \sum_{t \in T} \sum_{c \in C} p(t,c) log \left(\frac{p(t,c)}{p(t).p(c)} \right) \quad (2)$$

Where $I(T;C)$ denotes the mutual information between T and C, $T = \{0,1\}$ is the set in which a term t occurs ($T = 1$) or does not occur ($T = 0$) in a given document, and $C = \{0,1\}$ is the class set in which the document belongs to class c ($C = 1$), or does not belong to class c ($C = 0$)

Note that the size of the stoplists generated by both the MI and the TBRS methods is determined using the elbow approach as in the case of Z-Methods, i.e., ordering terms with respect to their informativeness values and search

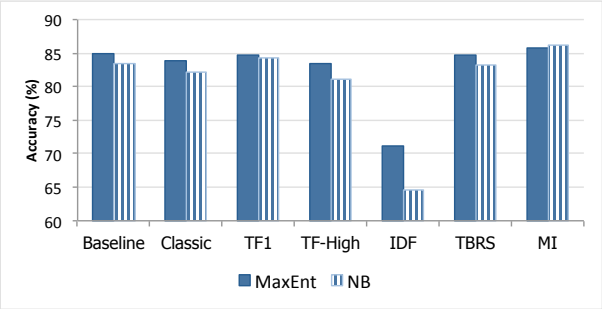


(a) Average Accuracy

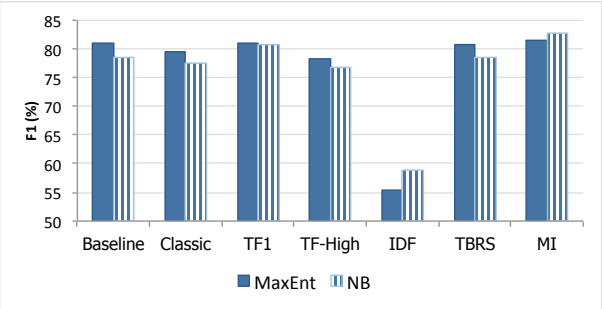


(b) Average F-measure

Figure 3: The baseline classification performance in Accuracy and F-measure of MaxEnt and NB classifiers across all datasets



(a) Average Accuracy



(b) Average F-measure

Figure 4: Average Accuracy and F-measure of MaxEnt and NB classifiers using different stoplists

for where the elbow appears in the rank-informativeness plot.

3.3. Twitter Sentiment Classifiers

To assess the effect of stopwords in sentiment classification we use two of the most popular supervised classifiers used in the literature of sentiment analysis, Maximum Entropy (MaxEnt) and Naive Bayes (NB) from Mallet.[1] We report the performance of both classifiers in accuracy and average F-measure using a 10-fold cross validation. Also, note that we use unigram features to train both classifiers in our experiments.

4. Experimental Results

To study the effect of stopword removal in Twitter sentiment classification we apply the previously described stopword removal methods and assess how they affect sentiment polarity classification (positive / negative classification of tweets). We assess the impact of removing stopwords by observing fluctuations (increases and decreases) on three different aspects of the sentiment classification task: the *classification performance*, measured in terms of accuracy and F-masure, the size of the classifier's *feature space* and the level of *data sparsity*. Our baseline for comparison is not removing stopwords.

Figure 3 shows the baseline classification performance in accuracy (a) and F-measure (b) for the MaxEnt and NB classifiers across all the datasets. As we can see, when no stopwords are removed, the MaxEnt classifier always outperforms the NB classifier in accuracy and F1 measure on all datasets.

4.1. Classification Performance

The first aspect that we study is how removing stopwords affects the classification performance. Figure 4 shows the average performances in accuracy (Figure 4:a) and F-measure (Figure 4:b) obtained from the MaxEnt and NB classifiers by using the previously described stopword removal methods. A similar performance trend can be observed for both classifiers. For example, a significant loss in accuracy and in F-measure is encountered when using the IDF stoplist, while the highest performance is always obtained when using the MI stoplist. It also worth noting that using the classic stoplist gives lower performance than the baseline with an average loss of 1.04% and 1.24% in accuracy and F-measure respectively. On the contrary, removing singleton words (the TF1 stoplist) improves the accuracy by 1.15% and F-measure by 2.65% compared to the classic stoplist. However, we notice that the TF1 stoplist gives 1.41% and 1.39% lower accuracy and F-measure than the MI stoplist respectively. Nonetheless, generating TF1 stoplists is much simpler than generating the MI ones in the sense that the former, as opposed to the latter, does not required any labelled data.

It can be also shown that removing the most frequent words (TF-Hight) hinders the average performance for both classifiers by 1.83% in accuracy and 2.12% in F-measure compared to the baseline. The TBRS stoplist seems to outperform the classic stoplist, but it just gives a similar performance to the baseline.

Finally, it seems that NB is more sensitive to removing

[1] `http://mallet.cs.umass.edu/`

stopwords than MaxEnt. NB faces more dramatic changes in accuracy than MaxEnt across the different stoplists. For example, compared with the baseline, the drop in accuracy in NB is noticeably higher than in MaxEnt when using the IDF stoplist.

4.2. Feature Space

The second aspect we study is the average reduction rate on the classifier's feature space caused by each of the studied stopword removal methods. Note that the size of the classifier's feature space is equivalent to the vocabulary size for the purpose of this study. As shown in Figure 5, removing singleton words reduces the feature space substantially by 65.24%. MI comes next with a reduction rate of 19.34%. On the other hand, removing the most frequent words (TF-High) has no actual effect on the feature space. All other stoplists reduces the number of features by less than 12%.

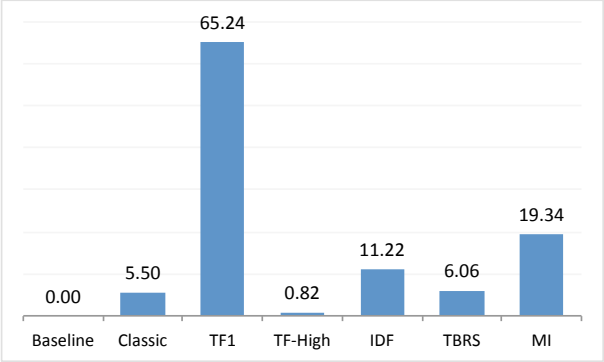


Figure 5: Reduction rate on the feature space of the various stoplists

Two-To-One Ratio As we have observed, removing singleton words reduces the feature space up to 65% on average. To understand what causes such a high reduction we analysed the number of singleton words in each dataset individually. As we can see in Figure 6 singleton words constitute two-thirds of the vocabulary size of all datasets. In other words, the ratio of singleton words to non singleton words is two to one for all datasets. This two-to-one ratio explains the large reduction rate in the feature space when removing singleton words.

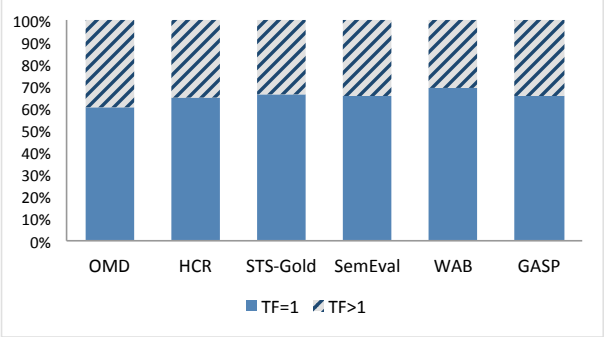


Figure 6: The number of singleton words to the number non singleton words in all datasets

4.3. Data Sparsity

Dataset sparsity is an important factor that affects the overall performance of a typical machine learning classifier (Phan et al., 2008). Saif et al. (2012a) showed that Twitter data are sparser than other types of data (e.g., movie review data) due to the large number of infrequent words present within tweets. Therefore, an important effect of a stoplist for Twitter sentiment analysis is to help in reducing the sparsity degree of the data.

To calculate the sparsity degree of a given dataset, we first construct the term-tweet matrix $G \in R^{m \times n}$, where m and n are the number of the unique terms (i.e., vocabulary size) and tweets in the dataset respectively. The value of an element $e_{i,j} \in G$ can be either 0 (i.e., the term i does not occur in tweet j) or 1 (i.e., the term i occurs in tweet j). According to the sparse nature of tweets data, matrix G will be mostly populated by *zero* elements.

The sparsity degree of G corresponds to the ratio between the number of the *zero* elements and total number of all elements (Makrehchi and Kamel, 2008) as follows:

$$S_d = \frac{\sum_j^n N_j}{n \times m} \quad (3)$$

Where N_j is the the number of *zero* elements in column j (i.e., tweet i). Here $S_d \in [0, 1]$, where high S_d values refer to high sparsity degree and vice versa.

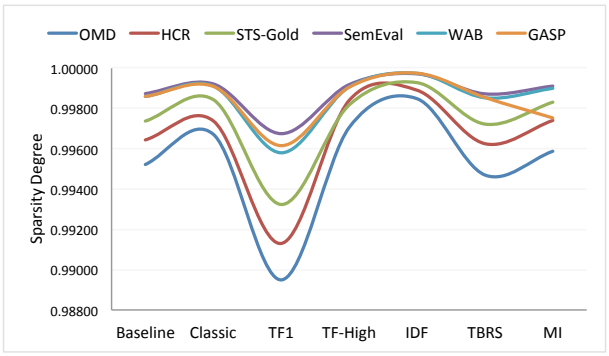


Figure 7: Stoplist impact on the sparsity degree of all datasets

Figure 7 illustrates the impact of the various stopword removal methods on the sparsity degree across the six datasets. We notice that our Twitter datasets are very sparse indeed, where the average sparsity degree of the baseline is 0.997. Compared to the baseline, using the TF1 method lowers the sparsity degree on all datasets by 0.37% on average. On the other hand, the effect of the TBRS stoplists is barely noticeable (less than 0.01% of reduction). It is also worth highlighting that all other stopword removal methods increase the sparsity effect with different degrees, including the classic, TF-High, IDF and MI.

From the above we notice that the reduction on the data sparsity caused by the TF1 method is moderate, although the reduction rate on the feature space is 65.24% as shown in the previous section. This is because removing singleton words reduces the number of *zero* elements in G as well as the total number of elements (i.e., $m \times n$) at very similar

Stoplist	Accuracy	F1	Reduction on Feature Space	Changes on Data Sparsity	Human Supervision Factor
Classic	83.09	78.46	5.50	0.08	No Supervision
TF1	84.50	80.85	**65.24**	**-0.37**	No Supervision
TF-High	82.31	77.58	0.82	0.1	Threshold Setup
IDF	67.85	57.07	11.22	0.182	Threshold Setup
TBRS	84.02	79.60	6.06	-0.017	Threshold Setup
MI	**85.91**	**82.23**	19.34	0.037	Threshold Setup and Data Annotation

Table 2: Average accuracy, F1, reduction rate on feature space and data sparsity of the six stoplist methods. Positive sparsity values refer to an increase in the sparsity degree while negative values refer to a decrease in the sparsity degree.

rates of 66.47% and 66.38% respectively as shown in Table 3. Therefore, the ratio in Equation 3 improves marginally, producing a small decrement in the sparsity degree.

Method	Reduction(Zero-Elm)	Reduction(All-Elm)
Calssic	3.398	3.452
TF1	66.469	66.382
TF-High	0.280	0.334
IDF	0.116	0.117
TBRS	3.427	3.424
MI	27.810	27.825

Table 3: Average reduction rate on *zero* elements (Zero-Elm) and all elements (All-Elm) of the six stoplist methods.

4.4. The Ideal Stoplist

The ideal stopword removal method is the one which helps maintaining a high classification performance, leads to shrinking the classifier's feature space and effectively reducing the data sparseness. Moreover, since Twitter operates in streaming fashion (i.e., millions of tweets are generated, sent and discarded instantly), the ideal stoplist method is required to have low runtime and storage complexity and to cope with the continuous shift in the sentiment class distribution in tweets. Lastly and most importantly, the human supervision factor (e.g., threshold setup, data annotation, manual validation, etc.) in the method's workflow should be minimal.

Table 2 sums up the evaluation results reported in the three previous subsections. In particular, it lists the average performances of the evaluated stoplist methods in terms of the sentiment classification accuracy and F-measure, reduction on the feature space and the data sparseness, and the type of the human supervision required. According to these results, the MI and the TF1 methods show very competitive performances comparing to other methods; the MI method comes first in accuracy and F1 measure while the TF1 method outperform all other methods in the amount of reduction on feature space and data sparseness.

Recalling Twitter's special steaming nature and looking at the human supervision factor, the TF1 method seems a simpler and more effective choice than the MI method. Firstly, because the notion behind TF1 is rather simple - "*stopwords are those which occur once in tweets*", and hence, the computational complexity of generating TF1 stoplists is generally low. Secondly, the TF1 method is fully unsupervised while the MI method needs two major human supervisions including: (i) deciding on the size of the generated stoplists, which is usually done empirically (See section 3.2.), and (ii) manually annotating tweet messages with their sentiment class label in order to calculate the informativeness values of terms as described in Equation 2.

Hence, in practical sentiment analysis applications on Twitter where a massive number of general tweets is required to be classified at high speed, a marginal loss in classification performance can be easily traded with simplicity, efficiency and low computational complexity. Therefore, the TF1 method is recommended for this type of applications. On the other hand, in applications where the data corpus consists of a small number of tweets of specific domain or topic, any gain in the classification performance is highly favoured over the other factors.

5. Discussion and Future Work

We evaluated the effectiveness of using several of-the-shelf stopword removing methods for polarity classification of tweets data. One factor that may affect our results is the choice of the sentiment classifier and the features used for classifier training. In this paper we used MaxEnt and NB classifiers trained from word unigrams. In future work, we plan to continue our evaluation with using other machine learning classifiers such as support vector machines and regression classifiers and training them from different set of features including word n-grams, part-of-speech tags, microblogging features (e.g., hashtags, emoticons, repeated letters), etc.

Our analysis revealed that the ratio of singleton words to non singleton words is two to one for all the six datasets. As described earlier, this ratio explains the high reduction rate on the feature space when removing singleton words. We aim to further investigate whether the two-to-one ratio can be generalised to any Twitter dataset. This can be done by randomly sampling a large amount of tweets over different periods of time and studying the consistency of our ratio along all data samples.

We focused our evaluation in this paper on Twitter data only. However, there are other microblogging services and social media platforms that have similar characteristics to Twitter. This includes Facebook, YouTube, MySpace and Tumblr. We aim in the future to conduct more experiments using data published on these platforms and investigate if we can obtain similar findings to those reported in this study.

6. Conclusions

In this paper we studied how six different stopword removal methods affect the sentiment polarity classification on Twit-

ter. Our observations indicated that, despite its popular use in Twitter sentiment analysis, the use of pre-compiled (classic) stoplist has a negative impact on the classification performance. We also observed that, although the MI stopword generation method obtains the best classification performance, it has a low impact on both the size of the feature space and the dataset sparsity degree.

A relevant conclusion of this study is that the TF1 stopword removal method is the one that obtains the best trade-off, reducing the feature space by nearly 65%, decreasing the data sparsity degree up to 0.37%, and maintaining a high classification performance. In practical applications for Twitter sentiment analysis, removing singleton words is the simplest, yet most effective practice, which keeps a very good trade-off between good performance and low processing time.

Finally, results showed that while the different stopword removal methods affect sentiment classifiers similarly, Naive Bayes classifiers are more sensitive to stopword removal than the Maximum Entropy ones.

7. Acknowledgements

This work was supported by the EU-FP7 project SENSE4US (grant no. 611242).

8. References

Asiaee T, A., Tepper, M., Banerjee, A., and Sapiro, G. (2012). If you are happy and you know it... tweet. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1602–1606. ACM.

Ayral, H. and Yavuz, S. (2011). An automated domain specific stop word generation method for natural language text classification. In *Innovations in Intelligent Systems and Applications (INISTA), 2011 International Symposium on*, pages 500–503. IEEE.

Bakliwal, A., Arora, P., Madhappan, S., Kapre, N., Singh, M., and Varma, V. (2012). Mining sentiments from tweets. *Proceedings of the WASSA*, 12.

Cover, T. M. and Thomas, J. A. (2012). *Elements of information theory*. John Wiley & Sons.

Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *The Journal of machine learning research*, 3:1289–1305.

Fox, C. (1992). Information retrieval data structures and algorithms. *Lexical Analysis and Stoplists*, pages 102–130.

Gokulakrishnan, B., Priyanthan, P., Ragavan, T., Prasath, N., and Perera, A. (2012). Opinion mining and sentiment analysis on a twitter data stream. In *Advances in ICT for Emerging Regions (ICTer), 2012 International Conference on*, pages 182–188. IEEE.

Hu, X., Tang, J., Gao, H., and Liu, H. (2013a). Unsupervised sentiment analysis with emotional signals. In *Proceedings of the 22nd international conference on World Wide Web*, pages 607–618. International World Wide Web Conferences Steering Committee.

Hu, X., Tang, L., Tang, J., and Liu, H. (2013b). Exploiting social relations for sentiment analysis in microblogging. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 537–546. ACM.

Kouloumpis, E., Wilson, T., and Moore, J. (2011). Twitter sentiment analysis: The good the bad and the omg! In *Proceedings of the ICWSM*, Barcelona, Spain.

Lo, R. T.-W., He, B., and Ounis, I. (2005). Automatically building a stopword list for an information retrieval system. In *Journal on Digital Information Management: Special Issue on the 5th Dutch-Belgian Information Retrieval Workshop (DIR)*.

Luhn, H. P. (1957). A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of research and development*, 1(4):309–317.

Makrehchi, M. and Kamel, M. S. (2008). Automatic extraction of domain-specific stopwords from labeled documents. In *Advances in information retrieval*, pages 222–233. Springer.

Martınez-Cámara, E., Montejo-Ráez, A., Martın-Valdivia, M., and Urena-López, L. (2013). Sinai: Machine learning and emotion of the crowd for sentiment analysis in microblogs. In *In Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013), Atlanta, Georgia, USA, June 2013.*

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in twitter. In *In Proceedings of the 7th International Workshop on Semantic Evaluation. Association for Computational Linguistics.*

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of LREC 2010*, Valletta, Malta.

Phan, X.-H., Nguyen, L.-M., and Horiguchi, S. (2008). Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th international conference on World Wide Web*, pages 91–100. ACM.

Rijsbergen, C. J. V. (1979). *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition.

Saif, H., He, Y., and Alani, H. (2012a). Alleviating data sparsity for twitter sentiment analysis. In *Proceedings, 2nd Workshop on Making Sense of Microposts (#MSM2012) in conjunction with WWW 2012*, Layon, France.

Saif, H., He, Y., and Alani, H. (2012b). Semantic sentiment analysis of twitter. In *Proceedings of the 11th international conference on The Semantic Web*, Boston, MA.

Saif, H., Fernandez, M., He, Y., and Alani, H. (2013). Evaluation datasets for twitter sentiment analysis. In *Proceedings, 1st Workshop on Emotion and Sentiment in Social and Expressive Media (ESSEM) in conjunction with AI*IA Conference*, Turin, Italy.

Shamma, D., Kennedy, L., and Churchill, E. (2009). Tweet the debates: understanding community annotation of uncollected sources. In *Proceedings of the first SIGMM workshop on Social media*, pages 3–10. ACM.

Silva, C. and Ribeiro, B. (2003). The importance of stop word removal on recall values in text categorization. In

Neural Networks, 2003. Proceedings of the International Joint Conference on, volume 3, pages 1661–1666. IEEE.

Sinka, M. P. and Corne, D. (2003a). Evolving better stoplists for document clustering and web intelligence. In *HIS*, pages 1015–1023.

Sinka, M. P. and Corne, D. W. (2003b). Towards modernised and web-specific stoplists for web document analysis. In *Web Intelligence, 2003. WI 2003. Proceedings. IEEE/WIC International Conference on*. IEEE.

Speriosu, M., Sudan, N., Upadhyay, S., and Baldridge, J. (2011). Twitter polarity classification with label propagation over lexical links and the follower graph. In *Proceedings of the EMNLP First workshop on Unsupervised Learning in NLP*, Edinburgh, Scotland.

Trumbach, C. C. and Payne, D. (2007). Identifying synonymous concepts in preparation for technology mining. *Journal of Information Science*, 33(6):660–677.

Xu, Y., Jones, G. J., Li, J., Wang, B., and Sun, C. (2007). A study on mutual information-based feature selection for text categorization. *Journal of Computational Information Systems*, 3(3):1007–1012.

Yang, Y. (1995). Noise reduction in a statistical approach to text categorization. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 256–263. ACM.

Zhang, L., Jia, Y., Zhou, B., and Han, Y. (2012). Microblogging sentiment analysis using emotional vector. In *Cloud and Green Computing (CGC), 2012 Second International Conference on*, pages 430–433. IEEE.

Zipf, G. K. (1949). Human behavior and the principle of least effort.