elty when these relationships change. We consider it likely that the structure of the neural network used will need to be evolved to give optimal performance in this task, and this process of evolution will need to take into consideration noise applied to the data.

References

[1] Markou, M., Singh, S. Novelty detection: a review part 2: neural network based approaches. Signal Process. 2003; 83:2499–2521

[2] Markou, M., Singh, S. Novelty detection: a review part 1: statistical approaches. Signal Process. 2003; 83:2481–2497

[3] Marsland, S., Nehmzow, U., Shapiro, J. Detecting novel features of an environment using habiutation. In: Proc. Simulation of Adaptive Behaviour, MIT Press, 2000, 189–198

[4] Marsland, S., Nehmzow, U., Shapiro, J. On-line novelty detection for autonomous mobile robots. Robo. Auto. Syst. 2005; 51:191–206

[5] Marsland, S., Shapiro, J., Nehmzow, U. A self-organising network that grows when required. Neural Netw. 2002; 15:1041–1058

[6] Jiang, J. Image compression with neural networks - a survey. Signal Process., Image Commun. 1999; 14:737–760

[7] Hosoya, T., Baccus, S.A., Meister, M. Dynamic predictive coding by the retina. Nature 2005; 436:71–77

[8] Stirling, P. Retina. In: Shepherd, G.M. (ed.) The Synaptic Organization of the Brain. 3rd edn. Oxford University Press, 1990, 170–213

[9] Mitchell, M. An Introduction to Genetic Algorithms. 6th edn. MIT Press, 1999

[10] Mitchell, T. Machine Learning. Int. edn. McGraw-Hill Higher Education, 1997

[11] Stanley, K.O. Efficient Evolution of Neural Networks Through Complexification. PhD thesis, University of Texas at Austin, 2004

[12] Whiteson, S., Stone, P., Stanley, K.O., Miikkulainen, R., Kohl, N. Automatic feature selection in neuroevolution. In: Proc. Genetic and Evolutionary Computation, ACM Press, 2005, 1225–1232

Selecting Bi-Tags for Sentiment Analysis of Text

Rahman Mukras, Nirmalie Wiratunga, and Robert Lothian

School of Computing
The Robert Gordon University
St Andrew Street
Aberdeen UK, AB25 1HG
{ram,nw,rml}@comp.rgu.ac.uk

Abstract. Sentiment Analysis aims to determine the overall sentiment orientation of a given input text. One motivation for research in this area is the need for consumer related industries to extract public opinion from online portals such as blogs, discussion boards, and reviews. Estimating sentiment orientation in text involves extraction of sentiment rich phrases and the aggregation of their sentiment orientation. Identifying sentiment rich phrases is typically achieved by using manually selected part-of-speech (PoS) patterns. In this paper we present an algorithm for automated discovery of PoS patterns from sentiment rich background data. Here PoS patterns are selected by applying standard feature selection heuristics: Information Gain (IG), Chi-Squared (CHI) score, and Document Frequency (DF). Experimental results from two real-world datasets suggests that classification accuracy is significantly better with DF selected patterns than with IG or the CHI score. Importantly, we also found DF selected patterns to result in comparative classifier accuracy to that of manually selected patterns.

1 Introduction

Sentiment Analysis involves the discovery and extraction of opinions contained in text. Recently this area has received much attention due to its potential applicability to lucrative industries such as marketing and the media. A typical application would be the classification of sentiments expressed within the widely uncharted domain of consumer generated media (e.g. online reviews and blogs). These domains play an increasingly important role in consumer related industries by providing direct and spontaneous feedback on public opinion [1, 11].

Much of the work in sentiment analysis has been devoted to the task of sentiment extraction by identifying subjective text [20]. Closely related to this is the need to establish intensity of extracted sentiment by measuring the deviation from non-subjective text [21]. A more constrained form of analysis involves the classification of text into two distinctive classes of positive and negative sentiment orientation [18, 12] In this paper we propose a part-of-speech (PoS) pattern selection algorithm to address the first problem area. A PoS pattern is composed of a sequence of consecutive PoS tags and is used to extract a set of phrases from an input text [8]. For example, the PoS pattern "JJ NN1" (an

adjective followed by a singular noun) can be used to extract bi-grams such as "fast car," "great person," or "evil motive." A bi-tag refers to a PoS pattern containing two consecutive PoS tags.

The algorithm we propose makes use of a background dataset to learn a set of bi-tags for extracting sentiment rich bi-grams. Each word in the background dataset is replaced with its respective PoS tag after which bi-tags are formed. Standard feature selection heuristics such as Information Gain (IG) [15, 22], Document Frequency (DF) [15, 22], and the Chi-Squared (CHI) score [22] are then applied to select the top discriminative bi-tags. Our hypothesis behind this is that bi-tags that are predictive of a particular sentiment orientation, should also extract bi-grams that are predictive of the same.

We evaluated the three feature selection heuristics on two test datasets by assessing their utility in sentiment classification, and found DF to yield best performances over IG and CHI. These results were contrary to what we expected, and were also in direct opposition to what is normally observed in text classification, where IG and CHI have traditionally been superior to DF [22, 15]. Given that both IG and CHI are designed to return relatively more discriminative bi-tags than DF [6], we speculate that these results indicate the absence of a one-to-one correspondence between the discriminative ability of a bi-tag, and the sentiment orientation of the bi-grams it extracts. Rather a useful bi-tag is one that occurs frequently across documents. We found also that the performance of DF is dependant on the availability of a sentiment-rich background dataset, whilst IG and CHI are unaffected by the choice of background data.

The remainder of this paper is organised as follows. Section 2 describes the process of selecting bi-tags for sentiment classification. Experimental results on two datasets are presented in Section 3. Related work appear in Section 4, followed by conclusions in Section 5.

2 Selecting Bi-Tags for Sentiment Classification

Fig. 1 illustrates a semisupervised approach to sentiment classification. Input text is tagged with corresponding PoS tags (we used the RASP PoS tagger[1] [2]). Bi-tags, obtained from a sentiment rich background dataset, are then used to extract sentiment rich bi-grams from the tagged text. The sentiment orientation of each bi-gram is then computed by comparing its association to two predefined sets of positive and negative words. These individual orientations are then aggregated to obtain the overall sentiment orientation of the input text.

Crucial to this semisupervised sentiment classification approach is the availability of bi-tags for bi-gram extraction. Existing approaches typically use manually selected bi-tags. Turney [18], for example, employs a set of manually crafted bi-tags similar to those listed in Table 1. In this Table, J refers to adjective forms (JJ, JJT, or JJR), NN1 and NN2 to a singular and plural nouns respectively, R to adverb forms (RR, RG, RGA, or RGR), and VV0 to a verb.

[1] Uses the CLAWS2 Tagset: http://www.comp.lancs.ac.uk/ucrel/claws2tags.html

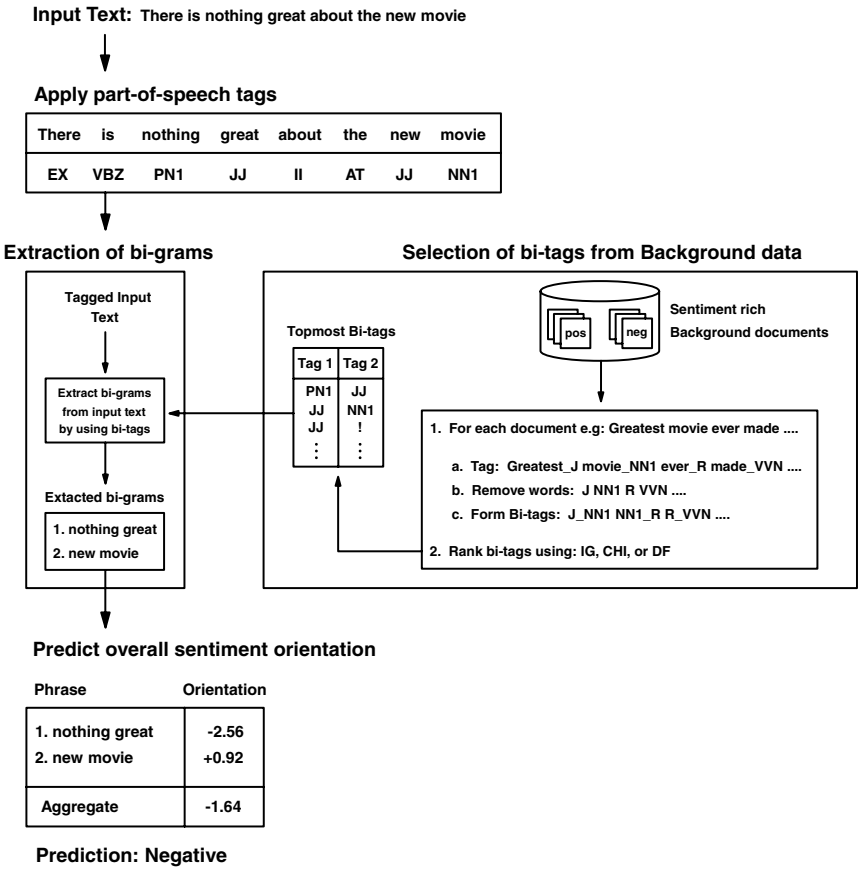

Fig. 1. Semisupervised Sentiment Classification.

To describe the patterns in Table 1, consider the fourth one which means that two consecutive words are extracted if the first is a noun and the second is an adjective, but the third (not extracted) cannot be a noun. The third word is checked so as to avoid extracting two consecutive bi-grams such as "very fast" and "fast car" from the initial phrase "very fast car." Note also that bi-grams are used instead of uni-grams so as to preserve context. For instance, "very good" and "not good" clearly posses opposing polarities. This information would be lost if we only use unigrams such as "good."

An obvious drawback of using manually selected bi-tags is that they need to be created by a domain expert in the first place. This can be a setback in practical applications such as blog-opinion filtering where maintenance of bi-tags is difficult. Consequently, in this paper we present an alternative procedure that automates the creation of bi-tags by use of a set of background documents.

Table 1. Manually Selected Bi-Tags.

	Tag 1	Tag 2	Tag 3 (Not Extracted)
1.	J	NN1 or NN2	anything
2.	R	J	not NN1 or NN2
3.	J	J	not NN1 or NN2
4.	NN1 or NN2	J	not NN1 or NN2
5.	R	VV0	anything

2.1 Selection of Bi-Tags from Background Data

As shown in Fig. 1, we use a sentiment rich bi-polar background dataset for bi-tag selection. A bi-polar dataset consists of documents belonging to either a sentiment positive or negative class. Each document in the background dataset is processed so that words are replaced by their PoS tag. Assuming that $t_1, t_2, \ldots, t_M$ is a sequence of PoS tags in an arbitrary document of this dataset, a bi-tag would be defined as $t_m t_{m+1}$ where $m = 1, 2, \ldots, M-1$. All such bi-tags are then ranked using a feature selection heuristic. In this study, we used Information Gain, Chi-Squared score, and document frequency.

Let q_k be the k^{th} bi-tag in the corpus, c^+ and c^- be the positive and negative classes respectively, and N be the total number of documents in the corpus. The following is how we implemented the above feature selection heuristics:

1. Information Gain:

$$IG(q_k) = \sum_{c \in \{c^+, c^-\}} \sum_{q \in \{q_k, \bar{q_k}\}} P(q, c) \log \frac{P(q, c)}{P(q)P(c)} \tag{1}$$

2. Chi-Squared score:

$$CHI(q_k) = \max_{c \in \{c^+, c^-\}} \left\{ \frac{N \cdot [P(q_k, c)P(\bar{q_k}, \bar{c}) - P(q_k, \bar{c})P(\bar{q_k}, c)]^2}{P(q_k)P(\bar{q_k})P(c)P(\bar{c})} \right\} \tag{2}$$

3. Document frequency:

$$DF(q_k) = N \cdot P(q_k) \tag{3}$$

We only recognise the presence of a bi-tag in a document when estimating the probabilities in Equations 1, 2, and 3. Once ranked, the topmost bi-tags are used to extract bi-grams from the input text. Table 2 illustrates, for each feature selection heuristic, a sample of the topmost bi-tags that were returned, along with a few of the bi-grams that they extracted.

Note that bi-grams extracted using DF such as "worst actor," and "terrible breakfast" are relatively more intuitive (in terms of sentiment richness) than the those extracted by IG and CHI such as "shrug ?" and "moron ?." We shall later show that classification performances also tend to follow the same trend.

Singular and plural proper nouns are avoided because they can adversely influence sentiment classification by being contextually associated with both positive and negative sentiments [18].

Table 2. Sample of top ranked bi-tags selected using IG CHI and DF.

Heuristic	Tag 1	Tag 2	Examples of extracted phrases
	NN1	?	shrug ?, glory ?, loser ?
	NN1	!	joke !, understatement !, menace !, perfection !
IG	VVZ	NN2	grate nerve, play scene, think woman
	NN2	NN2	work life, year job, concert movie
	J	VVG	good look, serious think, good fall
	NN1	J	guy worst, personality decent, spelling unattractive
	NN1	?	shrug ?, moron ?, fear ?, glory ?, loser ?
CHI	NN1	!	joke !, understatement !, menace !, perfection !
	NN1	VVZ	planet act, man pray, character play
	VV0	NN1	walk sunset, get sitcom, show emotion
	J	NN1	worst actor, terrible actress, worst breakfast
	NN1	NN1	example non-talent, quality style, courage range
DF	NN1	J	guy worst, personality decent, spelling unattractive
	NN1	NN2	going look, education work, world affair
	J	NN2	cute star, decent performance, outspoken topic

2.2 Predicting the Sentiment Orientation of the Input Text

Once the sentiment rich bi-grams have been extracted, then the next step is to compute their respective sentiment orientations. These orientations are later aggregated to compute the overall orientation of the input text.

Let b_i be the i^{th} extracted bi-gram from the input text. The sentiment orientation of b_i is computed by comparing its association to a set of positive words $\mathcal{P}$, against its association to a set of negative words $\mathcal{N}$ [18]. The words in these two sets are normally based on antonym pairs. For example, given an entry "good" in set $\mathcal{P}$, there would be a corresponding antonym such as "bad" in set $\mathcal{N}$.

Adjectives are known to be good carriers of sentiment [7], and therefore we compiled the two sets $\mathcal{P}$ and $\mathcal{N}$ from a list of manually selected adjectives as follows. For each word, we recorded an antonym set using a thesaurus, and a *familiarity score* using WordNet [5]. The familiarity score is a measure of a words usage in normal language. A high score would imply a commonly used word, whereas a low score would imply an uncommon word. This score is crucial in selecting the right words as computing association is difficult with either uncommon or excessively common words. We further augmented the familiarity score with word usage statistics obtained from a search engine.

Table 3 illustrates a sample of the list that we made. The two fields within the brackets of each word correspond to its familiarity score, and the number of hits it returned when queried in a search engine. Note that "good" would be unsuitable as it occurs too frequently. Similarly, "used" is also unsuitable due to its infrequent occurrence. The following are the words we finally chose:

$\mathcal{P}$ = {glad, rich, smart, great, wise, huge}
$\mathcal{N}$ = {sad, poor, stupid, terrible, foolish, little}

Table 3. A sample of the adjectives and their respective usage statistics.

Adjective Word	Corresponding Antonyms
new (11,1268194)	old (8,354828), used (3,3)
good (21,719768)	awful (6,29714), terrible (4,38042), bad (14,409)
general (6,574866)	special (7,195450)
right (14,549695)	wrong (9,180121), erroneous (1,2660)
great (6,514301)	terrible (4,38042), ordinary (2,28635)
big (13,410606)	small (10,248872), little (8,505147)
simple (7,245606)	complex (1,44198), difficult (2,77048)
poor (6,113213)	rich (12,74127)
huge (1,109800)	small (10,248872), little (8,505147)
glad (4,103213)	sad (3,82949), bittersweet (2,4273)
smart (7,86815)	stupid (3,104053), weak (12,28502)
foolish (2,10510)	wise (4,32497), all-knowing (1,0)

Association between two entities is computed using Pointwise Mutual Information [4] defined as follows:

$$I(x, y) = \log \left[\frac{P(x, y)}{P(x)P(y)} \right] \tag{4}$$

Here $I(x, y)$ has a minimum value of zero when x and y are independent of each other, and its value increases with the dependency between the two. This idea can be used to compute the sentiment orientation (SO) of b_i as follows,

$$SO(b_i) = I(b_i, \mathcal{P}) - I(b_i, \mathcal{N}) = \log \left[\frac{P(b_i, \mathcal{P})P(\mathcal{N})}{P(b_i, \mathcal{N})P(\mathcal{P})} \right] \tag{5}$$

Note that if b_i is equally associated to both $\mathcal{P}$ and $\mathcal{N}$, then $SO(b_i)$ would yield a value of zero. However, if b_i is more associated to either $\mathcal{P}$ or $\mathcal{N}$, then the value of $SO(b_i)$ would either be positive or negative respectively. To estimate the probabilities in Equation 5, we use the number of hits returned by a search engine given a query [18]. This was done as follows:

$$\begin{aligned} P(\mathcal{P}) &\simeq hits\,(\text{glad} \vee \ldots \vee \text{huge}) \\ P(\mathcal{N}) &\simeq hits\,(\text{sad} \vee \ldots \vee \text{little}) \\ P(b_i, \mathcal{P}) &\simeq hits\,(b_i \textbf{ near } (\text{glad} \vee \ldots \vee \text{huge})) \\ P(b_i, \mathcal{N}) &\simeq hits\,(b_i \textbf{ near } (\text{sad} \vee \ldots \vee \text{little})) \end{aligned}$$

Here $hits(\cdot)$ is a function that returns the number of documents that satisfy its query parameter, and **near** is a binary operator that constrains the search to documents containing its two query parameters, within 10 words of each other in any order (a similar approach was used in [18]). Finally the sentiment orientation of the input text is computed as the sign of the aggregate orientation

SEMANTIC-ORIENTATION$(d, Q, \mathcal{P}, \mathcal{N})$
1. $SO^d = 0$
2. $B \leftarrow$ EXTRACT-BIGRAMS(d, Q)
3. **for** each $b \in B$ **do**
4. $\quad SO^d += SO(b)$ // Accumulates the orientation of bi-grams.
5. **return** $sign\left[SO^d\right]$

EXTRACT-BIGRAMS(d, Q)
1. $B \leftarrow \{\}$
2. **for** $i = 1$ **to** $length(d) - 1$ **do** // $length(d)$ returns the number of words in d.
3. $\quad t = PoS(w_i^d w_{i+1}^d)$ // PoS returns the part-of-speech tags of the phrase.
4. $\quad$ **if** $t \in Q$ and $w_i^d w_{i+1}^d \notin B$ **then**
5. $\quad\quad B \leftarrow B \cup \{w_i^d w_{i+1}^d\}$
6. **return** B

Fig. 2. The Semantic Orientation Algorithm.

of its extracted bi-grams,

$$SO^d = sign\left[\sum_i SO(b_i)\right] \tag{6}$$

A positive aggregate would imply a positive orientation whereas a negative aggregate would imply a negative orientation. The algorithm is summarised in Fig. 2 whereby d is an input document consisting of all its words $w_1^d \ldots w_{length(d)}^d$, and Q is a set of bi-tags extracted using the approach discussed in Section 2.1.

3 Evaluation

The evaluation was performed on two Test datasets and two separate Background datasets. We also used the Trec Blog06 collection[2] [9] as a Query dataset to return the hits required to estimate the probabilities in Equation 5.

3.1 Datasets and Performance Metrics

The Test Datasets: Two bi-polar datasets were employed:

1. The Edmunds Dataset [10]: This dataset was composed of consumer reviews on used motor vehicles from the *Edmunds.com* website. Each review contained an ordinal label ranging from 1.0 to 9.8 step 0.2 (1.0 containing the most negative sentiment and 9.8 the most positive). Due to sparsity, we only used reviews within the range of 4.4 to 9.8. All reviews with less than 10 words were discarded and an equal class distribution was formed by randomly retaining 100 reviews per class. We then reduced the resultant into

[2] See http://ir.dcs.gla.ac.uk/test_collections/blog06info.html

Table 4. Comparison of bi-tag selection with IG, CHI and DF.

No of	Actors			Edmunds		
Patterns	IG	CHI	DF	IG	CHI	DF
1	0.49	0.50	**0.62**	0.50	0.50	**0.61**
2	0.49	0.55	**0.60**	0.50	0.51	0.62
3	0.49	0.55	**0.62**	0.50	0.51	0.59
4	0.52	0.55	**0.61**	0.51	0.51	0.59
5	0.53	0.57	**0.63**	0.52	0.52	0.60
6	0.54	0.57	**0.64**	0.52	0.54	0.59
7	0.53	0.57	**0.63**	0.51	0.53	0.60
8	0.53	0.57	**0.63**	0.50	0.53	0.59
9	0.52	0.57	0.62	0.50	0.50	**0.60**
10	0.53	0.57	**0.62**	0.50	0.50	**0.60**
11	0.53	0.57	**0.63**	0.50	0.50	**0.60**
12	0.53	0.58	0.62	0.50	0.54	**0.60**
13	0.52	0.59	0.62	0.51	0.55	**0.61**
14	0.53	0.59	**0.62**	0.52	0.55	0.58
15	0.53	0.60	0.62	0.52	0.55	0.58

bi-polar classes by assigning reviews labelled 4.4 and 4.6 to c^- and reviews labelled 9.6 and 9.8 to c^+.

2. The Actors Dataset [3]: This dataset was composed of reviews from the actors and actresses sub-topic of the *Rateitall.com* opinion website. Each review contained an ordinal integer label ranging from 1 to 5 (1 containing the most negative sentiment and 5 the most positive). All reviews that had less than 10 words were discarded. We restricted the number of reviews per author per rating to a maximum of 15, so as to avoid the bias of any prolific author from dominating the corpus [12]. We then reduced the resultant into bi-polar classes by assigning reviews labelled 1 to c^- and reviews labelled 5 to c^+. Finally, we formed an equal class distribution by randomly retaining 500 reviews per class.

Note that the formation of the bi-polar classes is sensible as classes at extremes of an ordinal scale possess opposite sentiment orientations and hence are essentially bi-polar. Both datasets were preprocessed using the sequence of tokenization, conversion to lowercase, PoS tagging, stemming, and finally stopword removal. We used a customised stopword list as we found that words such as "not," which is present in most stopword lists, to be quite useful in bi-grams such as "not good."

The Background Datasets: We employed a sentiment rich, and a non-sentiment-rich background dataset. For the former one, we used the Polarity dataset [12] which is composed of 2000 movie reviews (1000 positive reviews and 1000 negative ones). For the latter dataset, we used the ACQ, and EARN categories of the Reuters-21578 corpus [13]. These two categories are business

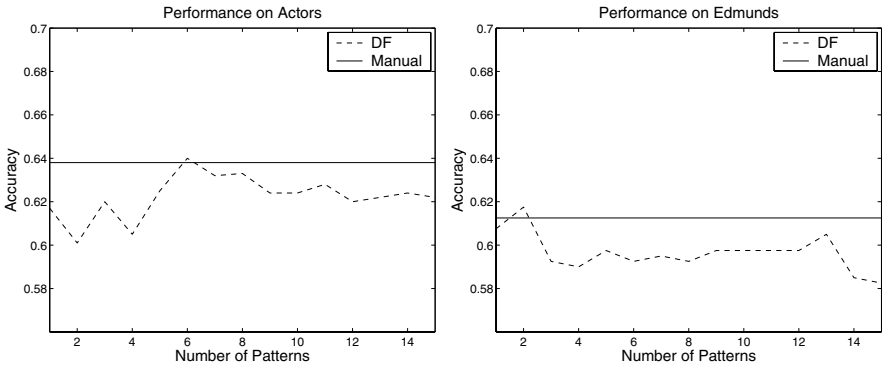

Fig. 3. Performance of manual and auto-generated bi-tags.

related and hence contain little, if any, sentiment rich information. We prepared this dataset by randomly selecting 1000 documents from each class such that each document belongs to at most one class. Both datasets were preprocessed in a fashion similar to the Test datasets.

The Query Dataset: We used the Trec Blog06 collection [9] to perform the queries that allowed us to realise the *hits*(.) function. This collection was compiled by the University of Glasgow and is composed of over 3.2 million blog posts. A blog post refers to an entry into a personal site that archives the posts in a reverse chronological order. Blogs are typically rich in opinion as they are authored by individuals who aim at expressing their opinions to the world. The Trec Blog06 collection was meant to be a realistic snapshot of the blogsphere (the collective term for all blogs), making it an excellent query dataset.

To prepare this collection, we extracted the text from the initial HTML format discarding all tokens that contained non-printable characters. We then preprocessed it using the sequence of tokenization, conversion to lowercase, stemming and stopword removal (with the specialised stopword list). We finally indexed the resultant collection, containing 2,466,650 documents, using the lucene[3] search engine.

Performance Metrics: Once bi-tags are learnt from the background dataset we used these to predict the sentiment orientation of unseen test data. The accuracy on test data is calculated for performance comparison. All reported results are the averages of 10 fold cross validation and significance is reported using the two-tailed paired t-test.

We also note that the semantic orientation of an input text can evaluate to zero. This often happens when extracted bi-grams are sparse in that they may

[3] http://lucene.apache.org

Table 5. Comparison of background data on the Actors dataset.

No of	IG		CHI		DF	
Patterns	SR	NSR	SR	NSR	SR	NSR
1	0.49	0.50	0.50	0.50	**0.62**	0.53
2	0.49	0.50	**0.55**	0.50	**0.60**	0.53
3	0.49	**0.52**	**0.55**	0.50	**0.62**	0.53
4	0.53	0.53	**0.55**	0.50	**0.61**	0.53
5	0.53	0.53	**0.57**	0.50	**0.63**	0.53
6	0.54	0.53	0.57	0.53	**0.64**	0.53
7	0.53	0.53	0.57	0.53	**0.63**	0.54
8	0.53	0.53	0.57	0.53	**0.63**	0.54
9	0.52	0.53	0.57	0.54	**0.62**	0.54
10	0.53	0.53	0.57	0.55	**0.62**	0.55
11	0.53	0.53	0.57	0.55	**0.63**	0.55
12	0.53	0.53	0.58	0.57	**0.62**	0.55
13	0.53	0.53	0.59	0.57	**0.62**	0.55
14	0.53	0.53	0.59	0.57	**0.62**	0.55
15	0.53	0.54	0.60	0.57	**0.62**	0.55

not co-occur with words in $\mathcal{P}$ or $\mathcal{N}$. It can also occur when no bi-grams are extracted from the input text. In such situations we chose the most commonly predicted class, and if this was a tie then we chose the positive class.

3.2 Experimental Results

We performed three main experiments: Firstly, we compared the three bi-tag selection heuristics (IG, CHI, and DF). Secondly, we compared the performance of automatically selected bi-tags against that of manually selected ones. Lastly, we assessed the effect of a non-sentiment-rich background dataset on performance.

Comparison of the Bi-Tag Selection Heuristics: Table 4 contains the classification accuracy achieved by the three bi-tag selection heuristics on the Actors and Edmunds datasets. Here we use the sentiment rich Polarity dataset as Background. Each row corresponds to results obtained with a particular PoS pattern size. For each row of each dataset, performances significantly better than the rest ($p < 0.05$) are shown in bold.

Note, in both datasets, that DF is on average better than both CHI and IG. Indeed this result was unexpected given the number of numerous studies that have reported the opposite trend in performance [22, 15]. Given that both IG and CHI are known to return relatively more discriminative bi-tags than DF [6], these results strongly suggest that the discriminative ability of a bi-tag does not directly influence that of the bi-grams it extracts.

Comparison with Manually Acquired Bi-Tags: Fig. 3 illustrates the results of comparing bi-tags selected from background data against manual bi-tags

Table 6. Comparison of background data on the Edmunds dataset.

No of	IG		CHI		DF	
Patterns	SR	NSR	SR	NSR	SR	NSR
1	**0.50**	0.47	0.50	0.51	**0.61**	0.55
2	**0.50**	0.47	0.51	0.47	**0.62**	0.55
3	0.50	0.51	0.51	0.47	**0.59**	0.55
4	0.51	0.50	0.51	0.48	0.59	0.55
5	0.52	0.50	0.52	0.47	0.60	0.55
6	0.52	0.50	0.54	0.53	0.59	0.55
7	0.51	0.50	0.53	0.52	**0.60**	0.53
8	0.50	0.49	0.53	0.52	**0.59**	0.53
9	0.50	0.50	0.50	0.53	**0.60**	0.52
10	0.50	0.50	0.50	0.52	**0.60**	0.54
11	0.50	0.50	0.50	0.50	**0.60**	0.54
12	0.50	0.49	0.54	0.53	**0.60**	0.54
13	0.51	0.49	0.55	0.53	**0.61**	0.54
14	0.52	0.48	0.55	0.53	**0.59**	0.53
15	0.52	0.52	0.55	0.54	**0.58**	0.53

shown in Table 1. Note, in this Figure, that the performance of manually selected bi-tags is independent of the x-axis and hence is a straight line.

Note that manual bi-tags perform, on average, better than bi-tags selected by DF. We, however, found this difference not to be statistically significant. These results were not unexpected as experience tells us that manual PoS pattern construction is rigorous and time-consuming. Each pattern, once derived, must be tested against a representative collection and fine tuned in light of the results. This is an iterative process that must be done by a domain expert. It is therefore not surprising to expect better performance when using such carefully designed patterns. However the advantage of the automatically generated bi-tags is that it reduces the demand on the knowledge engineer. This makes it suitable in applications whereby the data structure morphs rapidly making it infeasible to employ hand-crafted techniques.

Comparison with Non-Sentiment-Rich Background Data: We sought to investigate the role of a sentiment rich (SR) background dataset on the quality of generated bi-tags. To do this, we reran our experiments using the Reuters corpus as a non-sentiment-rich (NSR) background dataset and compared the results against the previous ones. Table 5 and 6 illustrate the results obtained on the Actors and Edmunds datasets. For each row of each feature selection heuristic, a significantly better ($p < 0.05$) performance is shown in bold.

Note, in both datasets, that DF performs significantly better on almost all pattern sizes when using a sentiment rich background dataset. This result strongly indicates the necessity of employing a sentiment rich background dataset to generate bi-tags. In contrast, there is almost no difference in the performances

of both CHI and IG on the two background datasets. This further supports our conclusion that the discriminative ability of a bi-tag does not necessarily translate to that of the bi-grams it extracts.

4 Related Work

We are aware of at least three other closely related efforts that focus on extracting sentiment rich information from text. Pang *et al* [12] noted that negation plays an important contextual role in identifying the sentiment orientation of text. For example, the word "not" in "not good" clearly flips the orientation of the word "good." To model this effect, Pang *et al* adapted a technique called negation tagging whereby a `NOT_` tag is added to every word between a negation word ("not," "isn't," "shouldn't," etc.), and the next punctuation mark. They found this procedure to be, on average, beneficial to sentiment classification performance. In contrast to our work, by extracting bi-grams rather than uni-grams, our system is by default capable of handling the problem of negation.

In another study, Rillof *et al* [14] employs PoS patterns to learn a dictionary of subjective nouns. Their algorithm starts with a set of patterns and ranks them based on their ability to extract a set of manually selected seed words. The approach is iterative in that, at each iteration patterns are ranked and the best once are carried over to the next iteration. The set of seed words is also updated with words extracted by the selected patterns at each iteration. A clear advantage of this method over ours is its iterative nature which provides it with the opportunity to incrementally refine the pattern set. However the bootstrapping method relies on the availability of a manually selected set of seed words at the start.

Finally, Turney [18] presents an unsupervised algorithm that classifies a review as *recommended* or *not recommended*. The algorithm performs the classification by computing the aggregate semantic orientation of a set of selected phrases extracted from the review [17, 16, 19]. The approach is similar to the one presented here except for the fact that it employs manually crafted bi-tags rather than mining them as we do. In Section 3.2 we found that comparative performance can be achieved with our approach. Importantly, the demand on the knowledge engineer is greatly reduced making it far more suited to dynamic environments, such as opinion filtering from blogs.

5 Conclusion

This paper presents a novel approach to PoS pattern selection for sentiment analysis of text. To the best of our knowledge, this is the first study in sentiment analysis that explores the possibility of applying feature selection heuristics to PoS pattern selection. Our approach achieves comparative performance against existing approaches that rely on manually selected PoS patterns.

An empirical evaluation of three bi-tag selection heuristics, showed DF to be the most effective over both IG and CHI. These results contradict previous work

on feature selection for text classification where IG and CHI have consistently outperformed DF [22]. Therefore we conclude that there exists a disparity between the sentiment orientation of a bi-tag and that of its extracted bi-grams. Instead, we find that bi-tags occurring frequently in a sentiment rich dataset, are good carriers of sentiment.

In future work we plan to extend the approach to accommodate a mixture of PoS pattern sizes. This would enable us to extract longer phrases such as "extremely superb vehicle," which occur frequently in sentiment rich text. We would also intend to improve the aggregation of extracted patterns when calculating sentiment orientation.

References

1. Lada Adamic and Natalie Glance. The Political Blogosphere and the 2004 U.S. Election: Divided They Blog. In *Proc. of 2nd Annual Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics*, 2005.
2. Ted Briscoe and John Carroll. Robust Accurate Statistical Annotation of General Text. In *Proc. of LREC*, pages 1499–1504, Las Palmas, Canary Islands, May 2002.
3. Sutanu Chakraborti, Rahman Mukras, Robert Lothian, Nirmalie Wiratunga, Stuart Watt, and David Harper. Supervised Latent Semantic Indexing using Adaptive Sprinkling. In *Proc. of IJCAI*, pages 1582–1587. AAAI Press, 2007.
4. Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, 1990.
5. Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.
6. George Forman. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. *JMLR*, 3:1289–1305, 2003.
7. Vasileios Hatzivassiloglou and Janyce M. Wiebe. Effects of Adjective Orientation and Gradability on Sentence Subjectivity. In *Proc. of Computational Linguistics*, pages 299–305, Morristown, NJ, USA, 2000. ACL.
8. John S. Justeson and Slava M. Katz. Technical Terminology: Some Linguistic Properties and an Algorithm for Identification in Text. *Natural Language Engineering*, 1:9–27, 1995.
9. Craig Macdonald and Iadh Ounis. The TREC Blogs06 Collection : Creating and Analysing a Blog Test Collection. Technical report, Department of Computing Science, University of Glasgow, Glasgow, UK, 2006.
10. Rahman Mukras, Nirmalie Wiratunga, Robert Lothian, Sutanu Chakraborti, and David Harper. Information Gain Feature Selection for Ordinal Text Classification using Probability Re-distribution. In *Proc. of IJCAI Textlink Workshop*, 2007.
11. Shinsuke Nakajima, Junichi Tatemura, Yoichiro Hino, Yoshinori Hara, and Katsumi Tanaka. Discovering Important Bloggers based on Analyzing Blog Threads. In *Proc. of 2nd Annual Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics*, 2005.
12. Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Proc. of EMNLP*, pages 79–86, 2002.
13. Reuters. Reuters-21578 text classification corpus. daviddlewis.com/resources/testcollections/reuters21578/, 1997.

14. Ellen Riloff, Janyce Wiebe, and Theresa Wilson. Learning Subjective Nouns using Extraction Pattern Bootstrapping. In *Proc. of CoNLL , ACL SIGNLL*, 2003.
15. Fabrizio Sebastiani. Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.
16. P.D. Turney and M.L. Littman. Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus. Technical report, National Research Council, Institute for Information Technology, 2002.
17. Peter D. Turney. Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In *Proc. of EMCL*, pages 491–502, London, UK, 2001. Springer-Verlag.
18. Peter D. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In *Proc. of ACL*, pages 417–424, Morristown, NJ, USA, 2002. ACL.
19. Peter D. Turney and Michael L. Littman. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Trans. Inf. Syst.*, 21(4):315–346, 2003.
20. Janyce Wiebe and Ellen Riloff. Creating Subjective and Objective Sentence Classifiers from Unannotated Texts. In *Proc. of CICLing*, pages 486–497, 2005.
21. Theresa Wilson, Janyce Wiebe, and Rebecca Hwa. Just How Mad Are You? Finding Strong and Weak Opinion Clauses. In *Proc. of AAAI*, pages 761–769. AAAI Press, 2004.
22. Yiming Yang and Jan O. Pedersen. A Comparative Study on Feature Selection in Text Categorization. In *Proc. of ICML*, pages 412–420. Morgan Kaufmann, 1997.