Natural Language Engineering **18** (1): 61–81. © Cambridge University Press 2011
doi:10.1017/S1351324911000118

A hierarchical approach to mood classification in blogs

FAZEL KESHTKAR and DIANA INKPEN
School of Information Technology and Engineering, University of Ottawa
Ottawa, Ontario, Canada
e-mail: {akeshta,diana}@site.uOttawa.ca

(*Received 9 September 2009; revised 22 April 2010; accepted 20 January 2011; first published online 11 March 2011*)

Abstract

In this article, we explore the task of mood classification for blog postings. We propose a novel approach that uses the hierarchy of possible moods to achieve better results than a standard machine learning approach. We also show that using sentiment orientation features improves the performance of classification. We used the Livejournal blog corpus as a data set to train and evaluate our method. We present extensive error analysis and discuss the difficulty of the task.

1 Introduction

Weblog services are free websites that allow people to write and submit their opinions and thoughts regarding social, political and other events. On some of these sites, users can also indicate their emotional state at the time of writing, by choosing a mood label. One of the most popular weblog services which offers this feature is Livejournal.[1] Livejournal provides 132 moods identifiers in a drop box list, and bloggers can use these to indicate their mood when they post a text or blog. Thus, they are labelling their posts based on a set of possible moods or emotions.

In this article, we address the task of classifying blog posts by mood. That is, given a blog post, we want to predict the most likely state of mind in which the post was written: whether the author was depressed, cheerful or bored and so on. As in the majority of text classification tasks, we take a machine learning approach, identifying a set of features to be used for the learning process.

Classifying texts by topic is a well-studied area Sebastiani 2002, as the words in the texts tend to be very good for characterizing topics. Classifying text by other stylistic properties is more difficult, since the documents can be about different topics in the same class or about the same topic in different classes.

Classifications of texts by expressed sentiments or emotions are examples of non-topical classification. Automatic classification of blogs by mood is a similar but more

[1] http://www.livejournalinc.com

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

challenging task, because of the colloquial language and because the texts tend to be short, which can causes progressive feature vectors used for classification. The differences between subjectivity/sentiment/emotion classification, and classification of texts by mood, are the number of classes and the type of distinctions that need to be learned from the texts. The two classes targeted in subjectivity classification are objective and subjective texts. In sentiment analysis, the classes are positive, negative and neutral. Several emotion classes are used in emotion analysis, while mood classification can include a very large number of classes.

A lot of progressive research has been done in opinion and sentiment analysis by Pang and Lee (2008). However, research on emotion and mood detection from texts is just beginning. Among the first preliminary investigations in emotion detection from texts are the small-scale experiments of Holzman and Pottenger (2003) and Rubin, Stanton and Liddy (2004).

A few researchers have studied mood classification in blogs. Mishne (2005) collected a corpus of blog data annotated with mood labels and then implemented a Support Vector Machine (SVM) classifier. The classification accuracy of this was only slightly above the baselines for both the training and test sets (the baseline values were 50 per cent because each set contained the same number of positive and negative examples).

Therefore, methods to increase the accuracy of the classification are required. Mishne (2005) used the top forty moods as classes, whereas we have used all 132 classes and their hierarchical organization provided by Livejournal. Users are not aware of this hierarchical relationship between moods when they choose one. The moods are not shown in an hierarchy drop-down menu, but in a flat drop-down list.

In this article, we introduce a hierarchical approach to mood classification. We address the variety of possible mood labels, and our approach is flexible in the sense that we can change the set of moods (i.e. the classes in which we classify the blogs). We decided to employ a hierarchical approach because we had such a large number of classes which were naturally grouped in a hierarchy. It makes sense for a classifier to learn the coarse-grained distinctions first and learn the more fine grained at subsequent levels in the hierarchy. We show that the results of the hierarchical classification are better than the results of a standard flat classification into 132 moods.

Our results also show that, although the blog corpus contains a large number of words in various domains, we can select features that lead to accurate classification of the expressed emotion and mood. We start with all the words as features (most of the words in the training data), then use feature selection techniques to reduce the feature space. We add sentiment orientation features and emoticons and show that they contribute significantly to improved classification.

The rest of the article is organized as follows. In Section 2, we describe the most recent related works. In Section 3, we examine the blog corpus that we consider for our research. In Section 4, we introduce our hierarchical classification approach. Section 4.2 follows with details regarding the features that we used for the classification process, divided into sets of related features. Our experiments and

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

results are reported in Section 5, and we compare our work to existing work in affect analysis in Section 6. Section 7 concludes the article.

2 Related work

In this section, we focus on research that has been done in mood classification divided into sets of related features to our research. In Section 6, we compare this research with our results.

2.1 Sentiment classification

In general, automatic sentiment classification can focus on words, sentences or documents. There are two approaches of sentiment classification methods for documents: lexicon based and corpus based. Lexicon-based methods compute a sentiment score for texts, according to the scores of the words in the texts that are also in a sentiment lexicon. Turney and Littman (2002) estimated the sentiment orientation of customer reviews by using the semantic orientation (SO) scores of the adjectives. The orientation of the adjectives was measured by their cooccurrence frequency on the web with several positive or negative seed adjectives. Kim and Hovy (2004) assigned polarity scores to a large list of words on the basis of their WordNet distance from positive and negative seed words.

Hiroshi *et al.* (2004) used deep language analysis techniques for machine translation to extract sentiment scores for words in documents. Kennedy and Inkpen (2006) determined the sentiment of customer reviews by counting positive and negative terms and taking into account contextual valence shifters, such as negations and intensifiers. A computable metric of positive or negative polarity in financial news text was proposed by Devitt and Ahmad (2007).

Corpus-based methods use a corpus of documents labelled with polarity to train a sentiment classifier. Various classification models and linguistic features were introduced, in order to improve the classification performance (Mullen and Collier 2004; Read 2005; Wilson, Wiebe and Hoffmann 2005; Pang and Lee 2008). McDonald *et al.* (2007) explored a structured model for jointly classifying the sentiment at different levels of granularity. Blitzer, Dredze and Pereira (2007) investigated domain adaptation for sentiment classifiers, focusing on online reviews for different types of products. Kennedy and Inkpen (2006) showed that combining classifiers with term counting led to improved results on product reviews. Andreevskaia and Bergler (2008) integrated a corpus-based classifier with a lexicon-based classifier, using precision-based weighting, in order to increase classification performance across different domains.

Several researchers focused on leveraging rich English resources for sentiment analysis in other languages. For the Romanian language, standard Naive Bayes and SVM classifiers have been applied for subjectivity classification (Banea *et al.* 2008), and the results show that automatic translation is viable for the construction of resources and tools in a new language. Wan (2008) used both Chinese and English lexicons to improve Chinese sentiment analysis. Work on Chinese sentiment

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

classification uses lexicon-based or corpus-based methods, similar to the methods described above for English (Tsou *et al.* 2005; Ye, Shi and Li 2006; Li and Sun 2007).

2.2 Manual identification of emotions in text

Quirk *et al.* (1985) have studied emotion in private states, in a wider framework. Another research on private states was done by Wiebe, Wilson and Cardie (2005). They selected 100,000 sentences of news articles and manually annotated them for emotions, opinions and sentiment (producing the multi- perspective question answering corpus). The Text Affect data set (Strapparava and Mihalcea 2007) was used in the SemEval 2007-Task 14. A development data set of 250 newspaper headlines was annotated. A test set of 1,000 newspaper headline annotations was released after the workshop. Strapparava and Mihalcea (2007) used six annotators. The agreement between the annotators (correlation) ranged from 0.36 to 0.68 for the six emotions annotated (including intensity levels). This work used the six classes of basic emotions proposed by Ekman (1992): *happiness*, *sadness*, *anger*, *disgust*, *surprise* and *fear*.

Aman and Szpakowicz (2007) produced 2,090 sentences annotated with the same six emotions, from 173 weblog posts. The interannotator agreement between the two judges ranged from 0.60 to 0.79 (kappa values) for different emotions.

In another research, the Affect Analysis Model was designed on the basis of the compositionality principle by Neviarouskaya, Prendinger and Ishizuka (2007). In this work, they used the nine basic emotions of Izard (1971): *anger*, *disgust*, *fear*, *guilt*, *interest*, *joy*, *sadness*, *shame* and *surprise*, as basis for affective text classification. Their rule-based algorithm enables analysis of emotions at different grammatical levels. Their system showed promising results in affect recognition in sentences extracted from diary-like blog posts and fairy tales. However, their system has limitations such as the strong dependency on the source lexicon and the affect database.

2.3 Automatic learning of emotion from texts

In the SemEval 2007-Task 14, the three teams that participated used unsupervised methods on the development set, with low accuracy of classification. More sophisticated unsupervised methods were proposed by Strapparava and Mihalcea (2008). Aman and Szpakowicz (2007), in their automatic classification experiments, achieved an accuracy of 73.89 per cent; they only focused on binary classification: whether a sentence contains an emotion or not.

Another research that focused on learning emotions from text is Alm, Roth and Sproat (2005). They have investigated automatic classification of sentences in children's fairy tales. They used the nine basic emotions of Izard (1971). They manually annotated with emotion information, a data set of stories. Read (2004) also has used a corpus of short stories, manually annotated with sentiment tags, in automatic emotion classification of sentences.

Liu, Lieberman and Selker (2003) have utilized real-world knowledge about affection from a common-sense knowledge base. In their research, they aim to learn

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Table 1. *Statistics about words and posts in the data set*

	Counts
Number of standard moods	132
Number of user-defined moods	54,487
Total words	69,149,217
Average words/post	200
Unique words	596,638
Individual pages	122,624
Total weblogs	37,009
Total posts	815,494

the semantics of text in order to identify emotions at the sentence level. They started with extracting from the knowledge base those sentences that contain some affective information. This information was used in building affective models of text, which were then used to label each sentence with a six tuple that corresponds to Ekman's six basic emotions.

Mihalcea and Liu (2006) have focused on two particular emotions: *happiness* and *sadness*. They worked on blog posts which are self-annotated by the blog writers with *happy* and *sad* mood labels.

There are a few researchers that have studied mood classification in blog corpora, such as Mishne (2005) and Jung, Park and Myaeng (2006). We explain these works in Section 6, then we compare their methods to ours.

3 Data set

We used the blog data set that Mishne (2005) collected for his research. The corpus contains 815,494 blog posts from Livejournal, a free weblog service used by millions of people to create weblogs. In Livejournal, users are able to optionally specify their current mood. To select their mood, users can choose from a list of 132 moods, or specify additional moods. We do not use the additional moods because very few posts are annotated with them. Some statistics of this corpus are shown in Table 1. From the total posts, only 77 per cent included an indication of the mood; we disregarded the rest. There are twenty-two posts per blog, on average. Table 2 shows the most frequent moods in the corpus, with their frequency counts. In the Livejournal weblog service, the moods are organized in a hierarchy, shown in Figure 1. Therefore, we will use this hierarchy to select the classes for each classification level, in the method that we propose in the next section.

4 Our hierarchical approach to mood classification

Researchers have been developing new techniques to classify large volumes of documents and texts (e.g. directories or biomedical data), particularly when the number of classes is large and the classes have a hierarchical structure (Dekel, Keshet and Singer 2004).

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Table 2. *The most frequent moods in corpus*

Mood	Occurrences	Mood	Occurrences
Amused	24,857 (4.0%)	Confused	8,160 (1.3%)
Tired	20,299 (3.2%)	Sick	7,848 (1.3%)
Happy	16,471 (2.6%)	Anxious	7,052 (1.1%)
Cheerful	12,979 (2.1%)	Axhausted	6,943 (1.1%)
Bored	12,757 (2.0%)	Crazy	6,433 (1.0%)
Accomplished	12,200 (1.9%)	Depressed	6,386 (1.0%)
Sleepy	11,565 (1.81%)	Curious	6,330 (1.0%)
Content	11,180 (1.8%)	Drained	6,260 (1.0%)
Excited	11,099 (1.8%)	Sad	6,128 (1.0%)
Contemplative	10,724 (1.65%)	Aggravated	5,967 (1.0%)
Contemplative	10,159 (1.61%)	Ecstatic	5,965 (1.0%)
Blah	10,127 (1.6%)	Blank	5,441 (0.98%)
Awake	10,121 (1.6%)	Hopeful	5,059 (0.95%)
Calm	10,052 (1.6%)	Thoughtful	4,295 (0.91%)
Bouncy	10,040 (1.6%)	Frustrated	4,132 (.90%)
Chipper	9,538 (1.5%)	Cranky	3,945 (0.85%)
Annoyed	8,277 (1.3%)	Loved	3,883 (0.65%)
Busy	7,956 (1.3%)	Angry	2,832 (0.60%)
Confused	7,724 (1.20%)	Working	2,775 (0.55%)
Annoyed	7,248 (1.15%)	Hyper	2,678(0.52%)

Hierarchical text categorization eliminates the assumption that categories are independent and instead tries to incorporate the interrelationship among the different categories. In a regular or flat text categorization approach, it is generally assumed that categories are independent and non-overlapping. This means that classification of an individual category is performed with the knowledge that all other categories are considered unrelated. In practice, however, categories tend to overlap and are not independent of each other (Wang, Zhou and Liew 1999).

Wang *et al.* (1999) introduced two types of hierarchical text categorization methods: *global* and *local*. We explain these techniques below and show how we applied hierarchical categorization techniques in the mood classification task.

- ***Global Approach:*** In the global approach only one classifier discriminates all categories in a hierarchy. It differs from a flat categorization by considering the relations between the categories (using a similarity measure).
- ***Local Approach:*** In the local approach, a classifier is built for each internal node in the hierarchy. It proceeds in a top–down fashion, starting with selecting the categories of the first level and then recursively choosing among the lower-level categories (the children of the higher-level categories). There are two possibilities. The first is to train one classifier for all the categories in the first level of the hierarchy (multiclass problem). Then one classifier is trained for each of the classes in the first level, to differentiate among all its subclasses. The process continues if there are more levels. The second

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

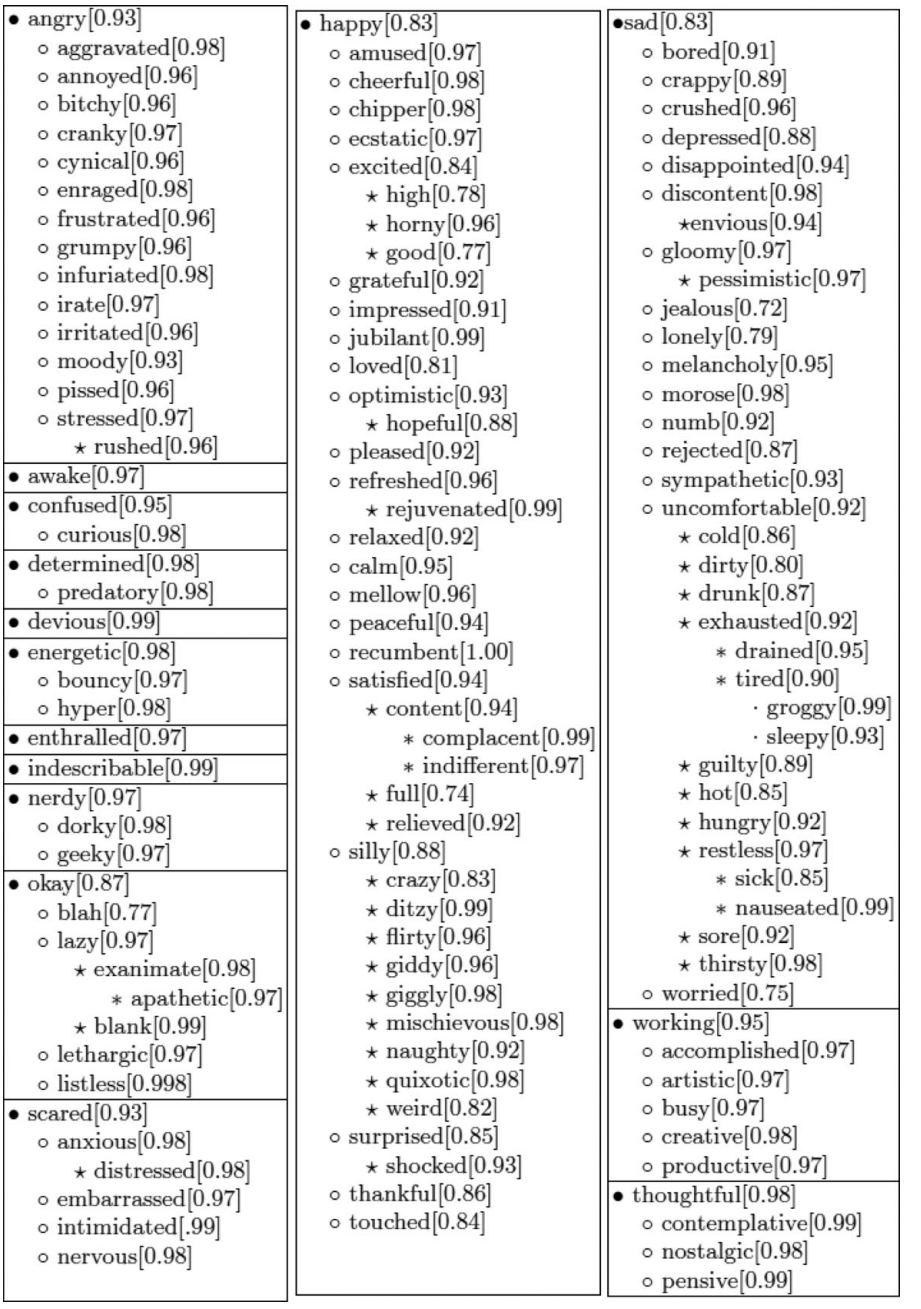


Fig. 1. The hierarchy of the 132 moods: •, Level 1; ○, Level 2; ⋆, Level 3; ∗, Level 4; and ·, Level 5. The numbers in the brackets indicate the best F-measure that we obtained for each node, as discussed later in Section 5.3.

possibility is to build a single-class classifier for each node in each level of the hierarchy (a binary classifier that differentiates the class of the node from any other sibling classes).

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

We used the first approach, with SVM classifiers, because it is more efficient. We had to choose between using *n*-ary classification at each node, or *n*-binary classifiers at each node (where *n* is the number of classes at each hierarchical level). We decided that only the *n*-binary classifiers method is feasible in our case: since we have a large number of classes the other would take too much time. Not only would the training times be too long but the testing time would as well when running the classifiers on new data. Moreover, we used SVM classifiers, which are normally binary; thus we used binary classification. The results were combined by the Weka tool, in order to produce the final *n*-ary classifier.

We used the local approach, and we tried both ways mentioned above. For the first one, we used *n*-ary decision trees. For the second one, we used SVM classifiers, which are binary by nature, and then we combined them into *n*-ary classifiers using the Sequential Minimization Optimization (SMO) algorithm (Keerthi *et al.* 2001) from Weka. We briefly describe this method for combining classifiers in Section 4.1.

The second approach led to much better results. This is why we report only the SMO results in Section 5.2. In general, this approach has the disadvantage that it requires a large number of classifiers, because there are five hierarchy levels and there are many subclasses in each node of the hierarchy; but the SMO method in particular does not have this disadvantage because it has an efficient way to combine the classifiers (without storing them as intermediary classifiers). Using the approach with other classifiers (e.g. decision tress) would make it very difficult to manually manage the many intermediary classifiers, and the results are not likely to be good, since, for this task, decision trees did not lead to good results.

4.1 Combining SVM classifiers

In the past few years, many researchers have become interested in SVM classifiers because they have shown significant generalization performance on a wide range of problems (Vapnik 1996; Burges 1998). The SMO algorithm is a fast implementation of SVM introduced by Platt (1998): tests have shown that SMO is often much faster and has better scaling properties than other SVM implementations. However, it has a significant deficiency, caused by the manner in which it maintains and updates a single threshold value. Keerthi *et al.* (2001) proposed a solution to this problem. They suggested using two threshold parameters and devised two modified versions of SMO that are more efficient than the original. Computational comparison on benchmark data sets showed that in most situations the modified versions perform significantly faster than the original SMO.

Another basic concern is that SVM is a binary classifier, and as such it cannot be used for problems with more than two classes. In the following, we examine the multiclass problem and the existing solutions.

The multiclass version of SMO uses the pairwise coupling procedure, which involves estimating class probabilities for each pair of classes, then coupling the estimates together. It starts by forming all the pairs of two classes, and training classifiers for them. Then all the pairwise decisions are combined to produce a multiclass classifier. The class that is finally chosen for a test instance is the class

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

that has the highest number of pairs. The thresholds for the pairs of classes are optimized simultaneously, based on the training data. For more details, see (Hastie and Tibshirani 1998).

In Weka, the implementation of Platt's SMO algorithm for training a support vector classifier uses the polynomial or the radial basis function kernel. The implementation globally replaces all missing values and transforms nominal attributes into binary ones. It also normalizes all attributes by default. Moreover, Weka's SMO is extended to the multiclass problem using the pairwise classification (Hastie and Tibshirani 1998) explained above. To obtain the best probability estimates, we used it with the option that fits logistic regression models to the outputs of the support vector machine. To show how Platt's SMO algorithm is used for classification in Weka, we show the Precision, Recall and F-measure for Level 1 of our hierarchal algorithm, later in the article, in Table 10. Also, Figure 3 illustrates the confusion matrix that Weka's SMO algorithm generates using binary classification for multiclass classification. Figure 3 shows the binary classification for the fifteen classes in Level 1. It compares each class with another class and shows the number of 'correct' and 'incorrect' instances for each class.

4.2 Feature set

A brief explanation of the features we used in our machine learning experiments follows. We used most of the features from Mishne (2005), plus some additional sentiment orientation features, such as tagged words from the General Inquirer (Stone *et al.* 2007).

4.2.1 Frequency counts

Bag-of-Words (BoW) is the most common feature representation used in automatic text classification. Most text classification systems consider a document as simply a 'bag of words' and use the frequency of the words as features (Sebastiani 2002); we also represented the words by their frequencies. Part-of-Speech tags are another common feature of text classification systems. We used the frequency of words and Part-of-Speech tag counts as features and TreeTagger (Schmid 1994) to extract the Part-of-Speech tags.

4.2.2 Length-related features

Since blog posts vary in length, we also considered length features, such as the length of the document, the number of sentences and the average number of words. We split the sentences using a simple method and used standard punctuation marks as sentence delimiters.

4.2.3 Sentiment orientation

For emotion and mood classification, the sentiment orientation of some words can be a useful feature for predicting emotion and mood. For example, some emotions

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Table 3. *Semtiment orientation values of words based on Kim and Hovy's list and on Turney and Littman's list*

Word	Kim–Hovy	Turney–Littman
Pricey	Positive	−4.99
Repetitive	Positive	−1.63
Teenage	Negative	−1.45
Momentary	Negative	+0.01
Fair	Positive	+0.02
Earnest	Positive	+1.86
Unparalleled	Negative	+3.67
Fortunate	Positive	+5.72

or moods, such as 'annoyed' and 'frustrated', are obviously *negative* while others, like 'cheerful' and 'loved', are clearly *positive*. So, it is expected that blogs posts with a positive aspect will, on average, have more positive orientation than negative blogs posts.

In our features set, we considered both the average word orientation and total orientation of a blog post as features. Determining the SO of a word is dependent on the method used for calculating it. We used several sources that are predictors for sentiment orientation. We calculated the total and the average orientation score for each document based on the words from the following resources:

- A list of 2,291 positive words and 1,915 negative words from the General Inquirer (Stone *et al.* 2007). These words were marked with tags such as active, passive, strong, weak, pleasure, pain, feeling (other than pain/pleasure), arousal, virtue, vice, overstated and understated.
- A list of 21,885 verbs and nouns that were automatically assigned a positive, negative, or neutral orientation score by Kim and Hovy (2004).
  They identified sentiments like the affective parts of opinions in their research. For this, they proposed a system that, for a given topic, automatically finds the people who hold opinions about that topic and the sentiment of each opinion. Their system contains a module for determining the sentiment of each word and a module for combining sentiments within a sentence. In their experiments, they used various models of classifying and combining sentiment at word and sentence levels, with promising results.
- A list of 1,718 adjectives with their scores of polarity values, constructed by using the method of Turney and Littman (2003). The method is based on measuring, for each word, its cooccurrence frequency on the web with a small set of positive adjectives and with a small set of negative adjectives.

Examples of words and their sentiment orientation scores in Kim and Hovy's list and in the adjectives list are shown in Table 3.

If there words had discrete values instead of numeric scores, we assigned a score of +1 to positive words and a score of −1 to negative words, and we calculated the sum and the average score of each text, for each of these three lists of words.

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

4.2.4 Special symbols

We used symbolic emotional icons (known as emoticons), which are printable characters intended to represent human emotions or attitudes. Emoticons are often horizontal textual representations of facial expressions (e.g. .:) = smile and ;= wink).

Today, emoticons are very popular in many domains, including blogs and chat rooms. The use of emoticons originated in email messages and quickly spread to other forms of online content. We used the nine most popular emoticons found in blog posts as features: tired, (: | ; smile, :); kiss, : −*; laughing, : *d*; sad, : (; love, : *x*; worried, : −*s*; surprise, : −*o*; angry, *x* − (.

5 Experiments and results

Our experiments and results are described in this section. We explain the performance of the simple flat classification into 132 moods and the performance of our hierarchy-based mood classification. We also explain the classification setting, then describe the experiments and results and investigate possible causes of errors.

5.1 Classification setting

We used the Weka data mining software (Witten and Frank 2005) for our experiments. We chose SVM as the main classifier because it has been shown to perform very well in text classification tasks, including previous studies on mood classification. Moreover, SVMs are able to deal with a large number of features and instances (Joachims 1998). We experimented with a few other classifiers, such as Nave Bayes and Decision Trees, but the results for our tasks were lower than those of SVM. As explained in Section 4.1, we used the SMO algorithm from Weka to combine binary SVM classifiers into multiclass classifiers.

Our feature space is very large, so for efficiency, we reduced the feature space for the BoW features by using feature selection methods. We used the *Chi-Square* feature selection method, in order to keep only the first 5,000 features from the 43,109 initial BoW features. In addition, we used the six sentiment orientation scores (for each of the three resources, the average score of the words from the text that are also in the resource and their total score) and the emoticons.

5.2 Experiments

Our training data set for the experiments includes 144,129 instances, which vary among the 132 different moods. From all the posts, we randomly selected the 144,129 as training data, as well as 90,000 as test data. Figure 2 shows the top twenty-nine moods in our experimental data set (the training and test data that we selected). The distribution is slightly different than the distribution in the whole LiveJournal data described in Table 2.

Our first experiment consisted of a flat classification into the 132 moods, using the SMO classifier. The second experiment was performed to evaluate the hierarchical classification method. For this purpose, we first trained a classifier to classify

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

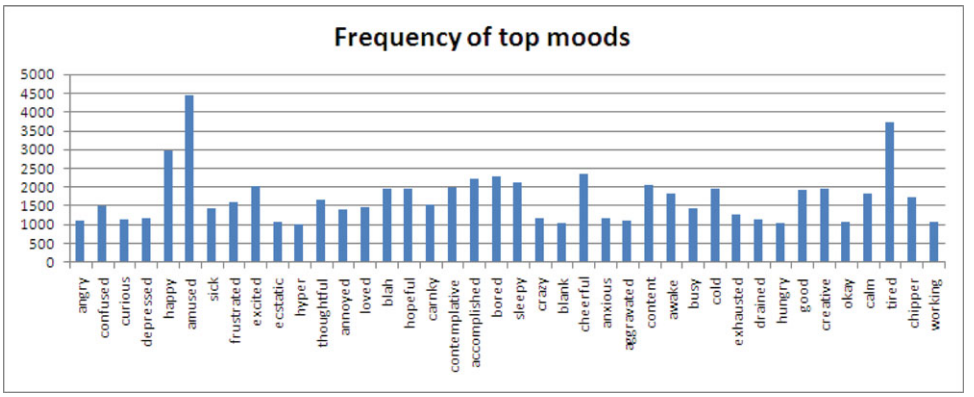


Fig. 2. Frequency of the top moods in the experimental data set.

into the fifteen categories from the first level of hierarchy (i.e. *happy, sad, angry, okay, working, scared, awake, thoughtful, nerdy, indescribable, enthralled, determined, confuse, devious* and *energetic*). In the next step, for each node from the first level of the hierarchy, we extracted the related instances and their mood labels. For example, for the node *angry* we selected all the documents that have the label *angry, aggravated, annoyed, bitchy, cranky, cynical, enraged, frustrated, grumpy, infuriated, irate, irritated, moody, pissed* and *stressed*. We then ran the classifier for each node in the second level. We repeated this procedure for each of the fifteen categories from the first level of the hierarchy and performed similar steps for the third, fourth and fifth level of the hierarchy.

For both sets of experiments above we ran our classifiers using BoW features and used BoW plus SO features (including emoticons). We found that adding the SO features greatly improves the classification results.

5.3 Results and discussion

The results of the first experiment, the flat classification into 132 moods, achieved an accuracy of 24.73 per cent for BoW+SO features and 18.29 per cent for *BoW* features. This is an improvement compared with a baseline accuracy result of 7 per cent when always choosing the most frequent mood (baseline is the lower expectation in each level for a naive classifier that always chooses the most frequent class).

For the second experiment, the hierarchical approach, the accuracy for the classifier that classifies into one of the fifteen moods from the first level was 95.87 per cent for BoW+SO and almost 40 per cent for BoW, compared with a baseline of 15 per cent; the results for Level 1 are illustrated in Table 4.

For the next step, we had fifteen potential classifiers in the second level, one for each node in the first level. We ultimately had eleven classifiers because four moods did not have any child branches in the hierarchy, so the classification for them was already completed. The average accuracy was 92.33 per cent for BoW+SO features, 83.30 per cent for BoW features only, and 32.70 per cent for a baseline of the most frequent class. The difference between the hierarchical approach with

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Table 4. *Accuracy for the hierarchical classification in Level 1 for both BoW and BoW+SO features*

	Baseline	BoW	BoW+SO
Level 1	15%	40%	95.85%

Table 5. *Accuracy for classification in Level 2 for both BoW and BoW+SO features*

Level 2	Baseline (%)	BoW(%)	BoW+SO(%)
Happy	8.64	62.72	86.97
Sad	10.38	66.89	86.88
Angry	11.67	80.13	91.90
Okay	24.55	78.67	82.25
Working	25.24	87.74	93.29
Scared	25.97	89.21	95.02
Thoughtful	35.99	91.32	94.84
Nerdy	41.40	90.65	97.68
Determined	65.52	93.25	95.83
Confused	56.32	85.71	94.33
Energetic	54.05	90.05	96.73
Average	32.70	83.30	92.33

the features and the baseline is 59.63 per cent. There are several branches that have fewer children and show larger improvement and several others with many children that show lower performance improvement. For example, the moods *happy*, *sad* and *angry* have many child branches and the improvement is smaller, while the gain in performance is bigger for moods such as *nerdy*, which has two branches. Two branches means three classes: in this case generic *nerdy* and two more specific types of *nerdy*: *geeky* and *dorky*. The results for level two are shown in Table 5.

The results of the Level 3 classifier are shown in Table 6 and the results of Level 4 in Table 7. Level 5 has only one classifier, for *tired*, with an accuracy of 96.22 per cent for BoW+SO features and 87.61 per cent for BoW, with a baseline of 54.44 per cent.

Our experiments and results clearly show that the hierarchical classification leads to strong performance and is well suited for the task. The summary of results for each level, shown in Table 9, supports this claim.

In Table 8, we added average precision, recall and F-measure values for each level, for the best hierarchical classification setting (BoW+SO). The best F-measure for each leaf node is also indicated in square brackets in Figure 1, in order to show results for each of the 132 moods.

To directly compare the results of the flat categorization to the results of the hierarchical classifiers, we cumulated the errors from all the levels. This provides a global accuracy of 79.92 per cent for all 132 classes for BoW+SO features and 18.29 per cent for the flat categorization. As illustrated in Table 9, the improvement

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Table 6. *Accuracy for classification in Level 3 for both BoW and BoW+SO features*

Level 3	Baseline(%)	BoW(%)	BoW+SO(%)
Uncomfortable	17.97	71.03	90.72
Surprised	56.18	96.19	97.68
Stressed	67.62	94.58	98.72
Silly	14.17	74.53	90.32
Satisfied	31.97	80.15	96.44
Refreshed	52.92	95.55	97.06
Optimistic	66.41	90.35	98.80
Lazy	31.37	87.40	96.88
Gloomy	66.21	93.68	99.88
Excited	35.87	86.33	91.75
Discontent	84.27	92.08	100
Anxious	58.88	89.91	96.85
Average	48.65	87.64	95.84

Table 7. *Accuracy for the hierarchical classification in Level 4 for both BoW and BoW+SO features*

Level 4	Baseline (%)	BoW (%)	BoW+SO (%)
Content	54.23	89.73	97.92
Restless	48.50	90.27	97.70
Exhausted	40.20	89.17	96.09
Exanimate	68.73	96.46	100
Average	52.16	91.40	97.93

in performance particularly especially when the sentiment orientation features are added. We believe the hierarchical method obtained better results because the classifiers are able to focus on a smaller number of classes and learn the distinctions between them.

Table 8. *Overall average results in each level*

Levels	Accuracy (%)	Precision	Recall	F-measure
Level 1	95.85	0.95	0.95	0.95
Level 2	92.33	0.92	0.92	0.91
Level 3	95.84	0.99	0.99	0.99
Level 4	97.93	1.00	0.97	0.99
Level 5	96.22	0.99	0.99	0.99

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Table 9. *The accuracy of the hierarchical classification when the classifiers from all the levels are applied successively (the errors from all the levels are multiplied), compared with the results of the flat classification, for both BoW and BoW+SO features*

Summary of the results	
Baseline	7.00%
Flat classification BoW	18.29%
Hierarchical classification BoW	23.65%
Flat classification BoW+SO	24.73%
Hierarchical classification BoW+SO	79.92%

5.4 Error analysis

The error rates are higher for classes with many subclasses in the hierarchy, which is likely because the subclasses are very similar to each other, so the algorithm cannot find enough features to distinguish between them. Another possible cause of errors is the nature of the hierarchy. We used it as it was provided, but there are a few areas where it does not seem logical, for example, the class 'silly' being a subclass of 'happy'. 'Shocked' and 'surprised' are also in the 'happy' class, when in reality they could be negative (as in pleasant surprise versus unpleasant surprise). In addition, some of the 132 possible classes are not well defined (e.g. recumbent and dorky), which could mean their labels have been arbitrarily chosen by the users who tagged their blogs.

To further investigate the difficulty of the task, we randomly selected a sample of blogs from those that were wrongly classified by our system. We wanted to see if human judges could classify them correctly (according to the tags chosen by the users who wrote the blogs) and, if not, determine the upper boundary of the accuracy that can be achieved by humans on this sample. An automatic system cannot be expected to produce better results than this upper boundary.

We cannot be certain that the data set was correctly annotated by the users who wrote the blogs. Some annotations might be wrong, which would lead to noise in the data set. We wanted to measure the agreement between human judges on this sample of data, to see how reliable the initial annotation is and how difficult the task is.

In general, the classification of opinions in texts is highly subjective, and this can cause disagreement between the annotations of human judges. Differences in the skills and focus of the judges, and ambiguity in both the annotation guidelines and the annotation task itself, also contribute to disagreement between judges (Passonneau 2006). We attempted to determine the degree of agreement between judges with respect to annotating blogs by mood; it is important to note that we did this by using a sample of difficult blogs. If we had selected the sample from blogs correctly classified by the system, it is likely that human judges would be able to classify them correctly, and their annotations would be in agreement.

From our classification task for the first level of hierarchy, we randomly selected 58 instances which were wrongly predicted by our system. For these instances, the

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Table 10. *Detailed results by class for Level 1*

Instances	Precision	Recall	F-measure	Class
1	0.975	0.984	0.98	Angry
2	0.957	0.991	0.974	Awake
3	0.925	0.982	0.952	Confused
4	0.98	0.984	0.982	Determined
5	0.993	0.997	0.995	Devious
6	0.985	1	0.992	Energetic
7	0.989	0.989	0.989	Enthralled
8	0.957	0.912	0.934	Happy
9	0.993	1	0.997	Indescribable
10	0.985	0.958	0.971	Nerdy
11	0.96	0.802	0.874	Okay
12	0.895	0.927	0.911	Sad
13	0.924	0.953	0.939	Scared
14	0.985	0.985	0.985	Thoughtful
15	0.878	0.917	0.897	Working
Weighted avg.	0.959	0.958	0.958	0.992

system was not able to predict the correct mood, as specified by the user of the LiveJournal. In order to provide more detailed results for Level 1 classification, we added precision, recall and F-measure for each subclass of this level (see Table 10) and a confusion matrix that shows the most frequent types of mistakes (see Figure 3). As explained in

We performed additional analysis on this random sample that our classifier had classified wrongly. We asked two human judges to assess this random sample by annotating each text with one of the moods in the first level of hierarchy (i.e. indescribable, confused, angry, awake, devious, determined, energetic, enthralled, happy, nerdy, okay, sad, scared, thoughtful and working). They were also allowed to specify a second label and to write comments.

We first measured the *correctness* of each judge. As shown in Figure 4, the highest correctness score for classification by human judges was 36.79 per cent that means the *upper boundary* for our system's correctness on this sample of data cannot be higher than 36.79 per cent. These results prove that the classification task in our blog corpus is very difficult. If we include the second choices of the judges as correct labelling, the highest correctness score goes up to 41.38 per cent.

We then measured the degree to which the judges agreed when classifying a document into a correspondent mood. The agreement between judges A and B was 65.52 per cent, and the *kappa* value that compensates for agreement by chance (Cohen 1960) was 0.1889. The low agreement and kappa values prove that the task is difficult, even for human annotators. Also, some of the initial tags in the data set cannot be trusted. According to the comments of the human judges, the usual cause of a wrong tag is the existence of multiple moods in the same blog. The writer of the blog could choose only one mood, and it was most likely that he/she would select

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Confusion Matrix for Level 1.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	<-- classified as
744	2	5	0	0	0	0	0	0	0	1	0	0	2	2	a = angry
0	629	1	1	0	1	1	0	0	1	0	0	0	1	0	b = awake
0	5	491	0	0	0	0	0	0	0	0	1	2	0	1	c = confused
2	1	1	437	0	1	0	0	0	0	0	1	0	1	0	d = determined
0	0	0	0	301	0	0	0	1	0	0	0	0	0	0	e = devious
0	0	0	0	0	392	0	0	0	0	0	0	0	0	0	f = energetic
0	0	1	0	1	0	261	0	0	0	0	0	1	0	0	g = enthralled
4	7	6	5	0	1	1	917	1	1	5	21	13	2	21	h = happy
0	0	0	0	0	0	0	0	444	0	0	0	0	0	0	i = indescribable
1	0	5	0	0	1	0	0	0	203	0	1	0	0	1	j = nerdy
2	4	5	0	0	1	0	26	0	1	287	11	3	0	18	k = okay
8	5	2	1	0	1	0	3	0	0	2	357	2	2	2	l = sad
0	0	8	0	0	0	0	4	0	0	1	1	306	0	1	m = scared
0	2	2	0	1	0	1	0	1	0	0	0	1	529	0	n = thoughtful
2	2	4	2	0	0	0	8	0	0	3	6	3	0	330	o = working

Fig. 3. Confusion matrix for correctly classified instances and incorrectly classified instances in Level l.

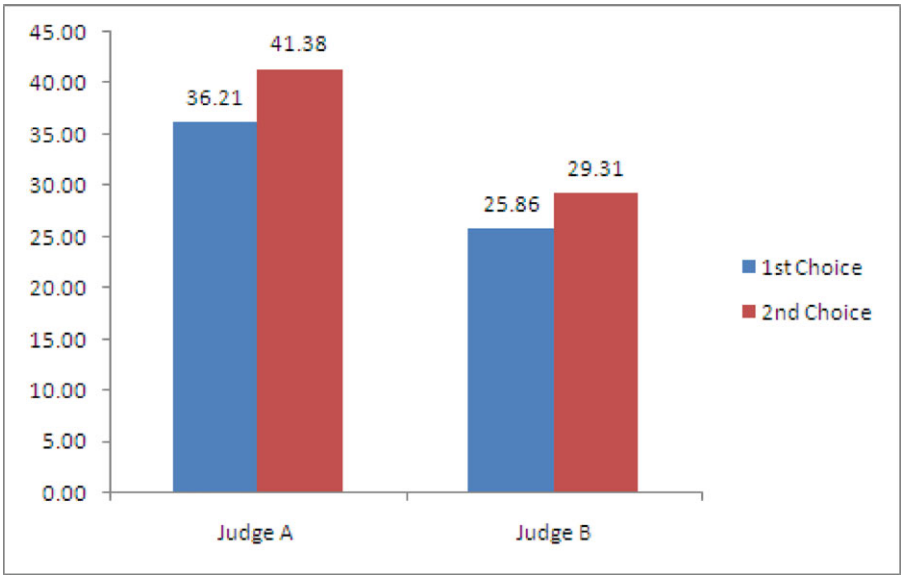


Fig. 4. The correctness results by two judges A and B.

the mood they were in when they started writing, then their mood changed during the writing, and this was reflected in the text. The judges also mentioned that some blogs described different moods of several persons that the blog author was writing about.

6 Comparison to related work

Standard research on text classification is intended to detect the topic of documents. Less research has been done to detect specific features in the text (Sebastiani 2002), such as the gender of the writer (Koppel, Argamon and Shimoni 2002) or authorship attribution (Koppel and Schler 2004).

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

In this article, we focused on mood classification. The task is difficult because of the distinctive aspects of mood in blogs. For example, a blogger can start writing in a certain mood, but the document can end with other content or mood. Some blogs are so intertwined that even human readers would have difficulty in identifying the mood and finding the relationship between the mood of the writer and the documents.

The work that is closest to our study is that of Mishne (2005), who used binary SVM classifiers for mood classification of blog data. As mentioned, we used the data set that he collected from LiveJournal. Mishne used the only forty most frequent moods, as shown in Table 2. He trained a classifier for each of the forty moods, on different training sets, then showed the performance of each mood, based on various sizes of the corresponding training set. In order to compare our work with Mishne's results, we also classified into the forty most frequent moods. We obtained 84.89 per cent accuracy, while the baseline given by the most frequent class is very low at 6.62 per cent and a random choice among forty classes results in 2.5 per cent. Mishne reported an average accuracy of 67 per cent. He used a test set randomly chosen for each mood with a balanced distribution of classes, which means a 50 per cent baseline for an approach that considers one class versus all the other classes. Therefore, we are unable to directly compare our results to his because we cannot use precisely the same test set, even though our test set is randomly chosen from the same data.

However, to directly compare our method to Mishne's algorithm, we implemented Mishne algorithm, for forty classes (the most frequent moods) in order to reproduce his results. To do so, we used our data set and we only kept Mishne's features (based on our best knowledge for Mishne's features set), then we trained an SVM classifier on our training data. We obtained an accuracy of 58.13 per cent for this classifier on our test set, on which we obtained the accuracy of 84.89 per cent by our algorithm. This shows that our algorithm performs 27.26 percentage points better than Mishne's algorithm for the top forty moods.

In summary, the essential differences between our work and Mishne's is that we use all the 132 moods, not only the forty most-frequent ones, and we enhanced the feature set with more sentiment orientation features. In order to achieve good results for such a large number of classes, we chose to use a different approach (i.e. hierarchical classification).

Other research that investigated mood classification was done by Jung *et al.* (2006) who proposed a hybrid approach to identify mood in blog texts. They used ConceptNet (Liu and Singh 2004) and the Affective Norms English Words (Bradley and Lang 1999) list. To perform classification, they chose unique features from blogs and calculated statistical features such as term frequency, n-grams, and pointwise mutual information (PMI) for the SVM classification method. They also used a paragraph-level segmentation based on mood flow analysis and extracted their own blog data from LiveJournal. They show good classification results only when restricting the task to four mood types: *happy*, *sad*, *angry* and *scared*. In addition, they considered only documents with lengths of sentences between four and forty.

Liu *et al.* (2003) introduced another approach that works on a larger corpus and is based on 'common-sense' classification. They used a corpus of 400,000 facts about

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

normal life in everyday life in the real world. They combined four linguistic models in order to recognize the affect. Their system analyzed the affective qualities of text, sentence by sentence. This has a practical value for people who want to evaluate the text they are writing.

Hierarchical categorization has been used for other text classification tasks. For example, Kiritchenko *et al.* (2006) applied this approach to biomedical texts, using a hierarchy from gene ontology, with good results. Wilson *et al.* (2005) applied a hierarchical approach to classification of customer reviews as positive, negative or neutral. They used a very shallow hierarchy, to classify into neutral and polar texts and to further classify the polar texts into positive and negative, with mixed results.

7 Conclusions and future work

This article presented experiments of hierarchy-based mood classification of blog corpus. In addition, a hierarchical approach was considered to classify the data using the SVM algorithm. Our corpus was collected from Livejournal, an online service that allows users to post their personal thoughts. Our results showed that the hierarchical approach provides substantial performance improvement over flat classification. Classifying mood in blog text is a difficult task, due to the variety of users. However, our approach shows that if we classify blogs by using the mood hierarchy, we can achieve very good performance. We also showed that using SO features on top of BoW features greatly improves the classification results.

In future work, we plan to experiment with more features sets, types of features and feature selection methods.

Acknowledgments

The authors wish to thank Soo-Min Kim and Ed Hovy for providing their polarity-tagged lists.

References

Alm, C. O., Roth, D., and Sproat, R. 2005. Emotions from text: lachine learning for text-based emotion prediction. In *Proceedings of HLT/EMNLP 2005*, pp. 579–86, Vancouver, British Columbia, Canada.

Aman, S., and Szpakowicz., S. 2007. Identifying expressions of emotion in text. In *Proceedings of Text, Speech and Dialog (TSD 2007)*, pp. 196–205, Prague, Czech Republic.

Andreevskaia, A., and Bergler, S. 2008. When specialists and generalists work together: overcoming domain dependence in sentiment tagging. In *Proceedings of ACL/HLT 2008*, Columbus, OH, USA.

Banea, C., Mihalcea R., Wiebe, J., and Hassan, S. 2008. Multilingual subjectivity analysis using machine translation. In *Proceedings of EMNLP 2008*, Waikiki, Honolulu, HI, USA.

Blitzer, J., Dredze, M., and Pereira, F. 2007. Biographies, Bollywood, boom-boxes and blenders: domain adaptation for sentiment classification. In *Proceedings of ACL 2007*, Prague, Czech Republic.

Bradley, M., and Lang, P. 1999. *Affective Norms for English Words*. University of Florida, Gainesville, FL, USA.

Burges, C. J. 1998. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery* **3**(2).

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Cohen, J. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* **20**: 37–46.

Dekel, O., Keshet, J., and Singer, Y. 2004. Large margin hierarchical classification. In *Proceedings of the 21st International Conference on Machine Learning*, Banff, Alberta, Canada.

Devitt, A., and Ahmad, K. 2007. Sentiment polarity identification in financial news: a cohesion-based approach. In *Proceedings of the ACL 2007*, Prague, Czech Republic.

Ekman, P. 1992. An argument for basic emotions. *Cognition and Emotion* **6**: 169–200.

Hastie, T., and Tibshirani, R. 1998. Classification by pairwise coupling. *The Annals of Statistics* **26**(2): 451–471.

Hiroshi, K., Tetsuya, N., and Hideo, W. 2004. Deeper sentiment analysis using machine translation technology. In *Proceedings of the COLING 2004*, University of Geneva, Switzerland.

Holzman, L., and Pottenger, W. 2003. Classification of emotions in internet chat: an application of machine learning using speech phonemes. Technical Report LU-CSE-03-002, Lehigh University.

Izard, C. E. 1971. *The Face of Emotion*. New York: Appleton-Century-Crofts.

Joachims, T. 1998. Text categorization with support vector machines: learning with many relevant features. In *Proceedings of ECML*, pp. 137–42, Chemnitz, Germany.

Jung, Y., Park, H., and Myaeng, S. 2006. A hybrid mood classification approach for blog text. *LNCS* **4099**: 1099–103.

Keerthi, S., Shevade, S., Bhattacharyya, C., and Murthy, K. 2001. Improvements to Platt's SMO algorithm for SVM classifier design. *Neural Computation* **13**(3): 637–49.

Kennedy, A., and Inkpen, D. 2006. Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence* **22**(2): 110–25.

Kim, S.-M., and Hovy, E. 2004. Determining the sentiment of opinions. In *Proceedings of COLING 2004*, University of Geneva, Switzerland.

Kiritchenko, S., Matwin, S., Nock, R., and Famili, F. 2006. Learning in the presence of class hierarchies. In *Proceedings of Learning@Snowbird 2006*, Snowbird UT, Princeton, NJ.

Koppel, M., Argamon, A., and Shimoni, A. 2002. Automatically categorizing written texts by author gender. *Literary and Linguistic Computing* **17**(4): 401–12.

Koppel, M., and Schler, J. 2004. Authorship verification as a one-class classification problem. In *Proceedings of the International Conference on Machine Learning (ICML 2004)*, Banff, Alberta, Canada.

Li, J., and Sun, M. 2007. Experimental study on sentiment classification of Chinese review using machine learning techniques. In *Proceedings of IEEE-NLPKE 2007*, Beijing, China.

Liu, H., Lieberman, H., and Selker, H. 2003. A model of textual affect sensing using real-world knowledge. In *Proceedings of the 8th International Conference on Intelligent User Interfaces (IUI 2003)*, pp. 125–32, Miami, FL, USA.

Liu, H., and Singh, P. 2004. Conceptnet - a practical commonsense reasoning tool-kit. *BT Technology Journal* **22**(4): 211–26.

McDonald, R., Hannan, K., Neylon, T., Wells, M., and Reynar, J. 2007. Structured models for fine-to-coarse sentiment analysis. In *Proceedings of ACL 2007*, Prague, Czech Republic.

Mihalcea, R., and Liu, H. 2006. A corpus-based approach to finding happiness. In *Proceedings of the AAAI Spring Symposium on Computational Approaches to Weblogs*, Stanford, CA.

Mishne, G. 2005. Experiments with mood classification in blog posts. In *Proceedings of ACM SIGIR 2005*, Salvador, Brazil.

Mullen, T., and Collier, N. 2004. Sentiment polarity identification in financial news: a cohesion-based approach. In *Proceedings of EMNLP 2004*, Barcelona, Spain.

Neviarouskaya, A., Prendinger, H., and Ishizuka, M. 2007. Analysis of affect expressed through the evolving language of online communication. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI 2007)*, pp. 278–81, Honolulu, HI.

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press

Pang, B., and Lee, L. 2008. Opinion minding and sentiment analysis. *Foundations and Trends in Information Retrieval* **2**(1–2): 1–135.

Passonneau, R. J. 2006. Measuring agreement on set-valued items (MASI) for semantic and pragmatic annotation. In *Proceeding of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy.

Platt, J. 1998. Fast training of support vector machines using sequential minimal optimization. In B. Schoelkopf, C. Burges, and A. Smola (eds.), *Advances in Kernel Methods, Support Vector Learning*. MIT Press, Cambridge, MA, USA.

Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J. 1985. *A Comprehensive Grammar of the English Language*. New York: Longman.

Read, J. 2004. *Recognizing Affect in Text Using Pointwise Mutual Information*. Master's thesis, Brighton, UK: University of Sussex.

Read, J. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of ACL 2005*, University of Michigan, Ann Arbor, MI.

Rubin, V., Stanton, J., and Liddy, E. 2004. Discerning emotions in texts. In *Proceedings of AAAI EAAT 2004*, Stanford University, Stanford, CA.

Schmid, H. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK.

Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM Computing Surveys* **34**(1): 1–47.

Stone, P. J., Dexter, D., Smith, C., Marshall, S., and Ogilvie, D. M. 2007. *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press, Cambridge, MA, USA.

Strapparava, C., and Mihalcea, R. 2007. SemEval-2007 Task 14: affective text. In *Proceedings of the 4th International Workshop on the Semantic Evaluations (SemEval 2007)*, Prague, Czech Republic, June 2007.

Strapparava, C., and Mihalcea, R. 2008. Learning to identify emotions in text. In *Proceedings of the ACM symposium on Applied Computing (SAC 2008)*, Sackville, New Brunswick, Canada, August 2008.

Tsou, B. K. Y., Yuen, R. W. M., Kwong, O. Y., La, T. B. Y., and Wong, W. L. 2005. Polarity classification of celebrity coverage in the Chinese press. In *Proceedings of International Conference on Intelligence Analysis*, McLean, VA, USA.

Turney, P., and Littman, M. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of ACL 2002*, Philadelphia, PA.

Turney, P., and Littman, M. 2003. Measuring praise and criticism: inference of semantic orientation from association. *ACM (TOIS)* **21**(4), pp. 315–346.

Vapnik, V. 1996. *The Nature of Statistical Learning Theory*. New York: Springer.

Wan, X. 2008. Using bilingual knowledge and ensemble techniques for unsupervised Chinese sentiment analysis. In *Proceedings of EMNLP 2008*, Honolulu, HI.

Wang, K., Zhou, S., and Liew, S. 1999. Building hierarchical classifiers using class proximities. In *Proceedings of VLDB 1999*, pp. 363–74, Edinburgh, Scotland, UK.

Wiebe, J., Wilson, T., and Cardie, C. 2005. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation* **39**(2–3): 165–210.

Wilson, T., Wiebe, J., and Hoffmann, P. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of HLT/EMNLP 2005*, Vancouver, British Columbia, Canada.

Witten, I., and Frank, E. 2005. *Data Mining: Practical Machine Learning Tools and Techniques*. 2nd ed. San Francisco: Morgan Kaufmann.

Ye, Q., Shi, W., and Li, Y. 2006. Sentiment classification for movie reviews in Chinese by improved semantic oriented approach. In *Proceedings of 39th Hawaii International Conference on System Sciences 2006*, Hawaii, Kauai, USA.

https://doi.org/10.1017/S1351324911000118 Published online by Cambridge University Press