Expert Systems with Applications 40 (2013) 621–633

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Document-level sentiment classification: An empirical comparison between SVM and ANN

Rodrigo Moraes, João Francisco Valiati *, Wilson P. Gavião Neto

Programa Interdisciplinar de Pós-Graduação em Computação Aplicada – PIPCA, Universidade do Vale do Rio dos Sinos – UNISINOS, Av. Unisinos, 950 São Leopoldo, RS, Brazil

ARTICLE INFO

Keywords:
Sentiment classification
Opinion mining
Text classification
Artificial Neural Networks
Support Vector Machines
Comparative study

ABSTRACT

Document-level sentiment classification aims to automate the task of classifying a textual review, which is given on a single topic, as expressing a positive or negative sentiment. In general, supervised methods consist of two stages: (i) extraction/selection of informative features and (ii) classification of reviews by using learning models like Support Vector Machines (SVM) and Naïve Bayes (NB). SVM have been extensively and successfully used as a sentiment learning approach while Artificial Neural Networks (ANN) have rarely been considered in comparative studies in the sentiment analysis literature. This paper presents an empirical comparison between SVM and ANN regarding document-level sentiment analysis. We discuss requirements, resulting models and contexts in which both approaches achieve better levels of classification accuracy. We adopt a standard evaluation context with popular supervised methods for feature selection and weighting in a traditional bag-of-words model. Except for some unbalanced data contexts, our experiments indicated that ANN produce superior or at least comparable results to SVM's. Specially on the benchmark dataset of Movies reviews, ANN outperformed SVM by a statistically significant difference, even on the context of unbalanced data. Our results have also confirmed some potential limitations of both models, which have been rarely discussed in the sentiment classification literature, like the computational cost of SVM at the running time and ANN at the training time.

1. Introduction

When consumers purchase products, a process of quality evaluation takes place naturally in their minds. From the industry point of view, to know the feelings among consumers may support strategic market decisions. On the other hand, potential consumers are often interested in the opinion of current customers in order to find out the choice that best fits their preferences. Nowadays, many people make their opinions available on the internet and researchers have been proposing methods to automate the task of classifying these textual reviews/opinions as positive or negative (Hatzivassiloglou & McKeown, 1997; Pang, Lee, & Vaithyanathan, 2002; Pang & Lee, 2008; Turney, 2002). The research field is know as *opinion mining* or *sentiment classification* and a complete overview on the subject is presented in Liu (2011) and Pang and Lee (2008). Basically, most of the methods in the literature are composed of two parts: (i) *feature selection* and (ii) *sentiment learning/classification* (Abbasi, France, Zhang, & Chen, 2011; Chen, Liu, & Chiu, 2011; Li, Xia, Zong, & Huang, 2009; Pang et al., 2002). In general, techniques in the literature can be differed in terms of the adopted approach for feature selection, since most of them agree on the learning techniques: Support Vector Machine (SVM) and Naïve Bayes (NB) (Abbasi, Chen, & Salem, 2008; Tang, Tan, & Cheng, 2009; Tsytsarau & Palpanas, 2012). As discussed in Section 2, Artificial Neural Networks (ANN) has attracted little attention as an approach for sentiment learning and a comparative study of ANN and popular sentiment classifiers under the same framework is still missing.

In this paper, we compare popular machine learning approaches (SVM and NB) with an ANN-based method in the context of document-level sentiment classification (Liu, 2011). In comparison with the sentiment classification literature, the main contributions of our work are: (i) a comparison of a dominant and a computationally efficient approach (SVM and NB, respectively) with an ANN-based approach under the same context; (ii) a comparison involving realistic contexts in which the ratio of positive and negative reviews is unbalanced; (iii) a performance evaluation of ANN on a full version of the benchmark dataset of Movies reviews (Pang & Lee, 2004). We are primarily interested in investigating the potential of an ANN-based approach for document-level sentiment classification and therefore we adopted classic supervised methods for feature selection and weighting in a traditional bag-of-words model (Manning, Raghavan, & Schtze, 2008).

This paper is organized as follows. In the next section we discuss the literature and justify the contributions of our work. In order to

* Corresponding author.
E-mail addresses: rodrigomorae@gmail.com (R. Moraes), jfvaliati@unisinos.br (J.F. Valiati), wgaviao@gmail.com (W.P. Gavião Neto).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.07.059

approach a standard comparative context, Section 3 presents an overview of usual techniques in sentiment analysis. Section 4 presents an overview of NB, SVM and ANN approaches as well as a fundamental comparison between ANN and SVM. The set-up of our experiments is reported in Section 5 and the results are discussed in Sections 6 and 7. Section 8 summarizes our conclusions.

2. Literature review

Many researchers have been addressing the problem of sentiment classification on textual reviews. Some datasets are available and have been used by many researchers in order to compare results, and the dataset of movie reviews (Pang & Lee, 2004) is the most popular benchmark dataset in the literature.[1] Since the focus of our study is on the overall opinion (positive or negative) expressed in a review, we have oriented our literature review towards document-level sentiment classification, which assumes that a review document expresses opinions on a single product or service and was written by a single reviewer/customer.

2.1. Document-level sentiment classification

Although some approaches have proposed using unsupervised/semi-supervised learning methods (Lin & He, 2009; Turney, 2002), most of the work has focused on supervised learning techniques. Pang et al. (2002) proposed a seminal approach in sentiment classification. Essentially, they conclude that machine learning techniques, like NB and SVM, do not achieve an accuracy as good on sentiment classification as on traditional topic-based categorization. In order to better approach sentiments in textual reviews, Pang and Lee (2004) also proposed classifying sentences as being either subjective or objective, and then apply sentiment classification on the subjective portion of the text. In addition to the probability of a term being subjective, Raychev and Nakov (2009) proposed considering the position of terms in the text as a strategy for identifying informative features.

Whitelaw, Garg, and Argamon (2005) proposed considering adjectival expressions as an important indication of the sentiment polarity in textual reviews. Zaidan, Eisner, and Piatko (2007) proposed learning the sentiment polarity of reviews from an additional source of information. Basically, human annotators were requested to highlight the most important words and sentences that justify why a review is positive or negative. Li et al. (2009) analyzed six popular feature selection methods and conclude that terms with (i) higher *document frequency* and (ii) higher *category ratio* are more informative/effective for classification. O'Keefe and Koprinska (2009) also evaluated feature selection techniques as well as feature weighting methods. Best results were achieved using the *categorical proportional difference* as a feature selection metric, which is close in meaning to the *category ratio* discussed in Li et al. (2009).

Abbasi (2010) proposed that feature selection methods should be tailored to sentiment analysis by combining syntactic properties of text features with sentiment-related semantic information. Syntactic information is computed by considering rule-based relations between various categories of features, like simple words and part-of-speech tags. Semantic information is computed by assigning weight to features according to their (i) occurrence distributions across categories in the training data and (ii) degree of subjectivity, which is derived from SentiWordNet (Baccianella & Sebastiani, 2010), a publicly available lexical resource that contains sentiment polarity scores. Dang, Zhang, and Chen (2010) also used various categories of features, which were refined by applying the *Information Gain* (IG) technique (Yang & Pedersen, 1997). Authors proposed a process of extracting sentiment features, what involves identifying adjectives, adverbs and verbs, and assigning sentiment scores to them according to the SentiWordNet. Recently, He, Lin, and Alani (2011) proposed to detect sentiment and topic simultaneously from text and show that a state-of-the-art performance can be achieved by augmenting features with polarity word labels.

As discussed above, literature is focused on feature selection, which provides the input to sentiment learning algorithms. The most popular sentiment learning techniques are SVM and NB, and many authors have reported better accuracy by using SVM (Abbasi, 2010; Dang et al., 2010; O'Keefe & Koprinska, 2009; Pang et al., 2002; Prabowo & Thelwall, 2009; Ye, Zhang, & Law, 2009).

Another characteristic of the sentiment classification literature is that many methods have been tested only on balanced datasets and there has been little discussion on the effects of learning subjective aspects from unbalanced data, although it is typical of the product domain to have substantially more positive than negative reviews (Burns, Bi, Wang, & Anderson, 2011; Li, Wang, Zhou, & Lee, 2011a). Burns et al. (2011) address sentiment classification on unbalanced data, however the experiments do not involve neither SVM nor ANN. Li et al. (2011a, 2011b) adopt a random under-sampling method, which is a popular approach to deal with imbalanced data. The major drawback of random under-sampling is that it can discard potentially useful data that could be important for the learning process. In order to overcome this problem, Wang, Li, Zhou, Li, and Zhu (2011) propose combining multiple classifiers, which are trained from multiple instances of under-sampled data. This ensemble learning approach can be computationally expensive (Xia, Zong, & Li, 2011) and no discussion on this issue is reported by the authors. In contrast to those approaches that discuss imbalanced sentiment classification, we evaluate the performance of ANN as the learning approach and report results in a context in which no previous stages are considered to deal with imbalanced data, like sampling techniques (Van Hulse, Khoshgoftaar, & Napolitano, 2007).

2.2. ANN and sentiment classification

ANN has figured rarely in the literature (Bespalov, Bai, Qi, & Shokoufandeh, 2011; Chen et al., 2011; Claster, Hung, & Shanmuganathan, 2010; Zhu, XU, & shi Wang, 2010), as can be seen in recent surveys on sentiment analysis (Pang & Lee, 2008; Tsytsarau & Palpanas, 2012). In order to reduce the training time, Chen et al. (2011) propose combining word features to model an ANN-based approach with few input neurons. However, the experiments do not involve unbalanced datasets as well as a comparison with popular sentiment learning techniques like NB and SVM. Zhu et al. (2010) propose simulating the human's judgment on sentiment polarity by using an ANN-based individual model. Authors report a comparative analysis between the proposed ANN-based method and SVM, however a performance evaluation on unbalanced datasets is not discussed. Instead of classifying positive versus negative reviews, Claster et al. (2010) propose using self-organizing maps (SOM) to cluster microblog posts according to subjective aspects in Movies domain, like "funny" and "predictable". Perhaps the most conclusive experiments that compare variants of an ANN-based method with SVM-based approaches for sentiment learning are reported in Bespalov et al. (2011). However, authors considered only balanced data in the experiments and did not discuss computational issues, hence our work can be understood as an extension of Bespalov et al. (2011) to the context of unbalanced datasets as well as in discussing the computational requirements of both ANN and SVM models to achieve comparable results.

[1] See <http://www.cs.cornell.edu/people/pabo/movie-review-data/otherexperiments.html>.

3. Background and usual techniques

In this section, we present an overview of steps and techniques commonly used in sentiment classification approaches, as shown in Fig. 1. We follow the popular *bag-of-words* model in which a document is represented as a vector, whose entries correspond to individual terms of a vocabulary.

Pre-processing techniques are often used to remove *stopwords*, which are common terms like prepositions and articles, and reduce term variations to a single representation by applying *stemming* procedures (Weiss, Indurkhya, & Zhang, 2004). Popular *stemmer* algorithms for the english language are Snowball (Porter, 2001), Porter (Porter, 1980) and Lovins (Lovins, 1968).

Next, a numerical representation is computed from textual data. *Binary* representation is widely used and only takes into account presence or absence of a term in a document. The number of times a term occurs in a document (i.e., *term frequency*) is also used as a weighting scheme for textual data (Li et al., 2009; Paltoglou & Thelwall, 2010). TF-IDF (*Term Frequency – Inverse Document Frequency*) is one of the most popular representations and considers not only term frequencies in a document, but also the relevance of a term in the entire collection of documents. The classic TF-IDF$_{t,d}$ (Manning et al., 2008) assigns to term t a weight in document d as

$$\mathrm{TF-IDF}_{t,d} = \mathrm{TF}_{t,d} \times \mathrm{IDF}_t, \quad \text{where } \mathrm{IDF}_t = \log \frac{N}{\mathrm{DF}_t}. \tag{1}$$

$\mathrm{TF}_{t,d}$ is the number of occurrences of term t in document d, N is the number of documents in the collection and DF_t is the number of documents in the collection that contain term t. Essentially, TF-IDF avoids assigning high scores to terms that occur too often in the dataset.

Another stage commonly found in sentiment classification approaches is feature selection. It can make classifiers more efficient/effective by reducing the amount of data to be analyzed as well as identifying relevant features to be considered in the learning process. Usual feature selection methods are *document frequency* (Bai, 2011; Dang et al., 2010; Pang et al., 2002), *mutual information* (Li et al., 2009; Turney, 2002), *information gain* (Abbasi et al., 2011, 2008; Li et al., 2009; Riloff, Patwardhan, & Wiebe, 2006) and *chi-square* (Abbasi et al., 2011; Li et al., 2009). None of them has been widely accepted as the best feature selection method for sentiment classification or text categorization, however, information gain has often been competitive (Abbasi et al., 2011; Forman, 2003; Li et al., 2009; Xia & Zong, 2010; Yang & Pedersen, 1997). It ranks terms by considering their presence and absence in each class (Berry & Kogan, 2010). A high score is assigned to terms that occur frequently in a class (and rarely in the others) as follows (Weiss, Indurkhya, & Zhang, 2010):

$$IG(t) = \sum_{k=1}^{C} P(c_k) \log \frac{1}{P(c_k)} - \sum_{t \in \{t_p, \overline{t_p}\}} P(t) \sum_{k=1}^{C} P(t|c_k) \log \frac{1}{P(t|c_k)}, \tag{2}$$

where $P(c_k)$ is the prior probability of a document occurring in class c_k, $P(t)$ is the probability of term t occurring or not in a document, i.e. $P(t_p)$ and $P(\overline{t_p})$ respectively. $P(t|c_k)$ is the conditional probability of term t occurring or not in a document of class c_k and C is the number of classes.

Ideally, the feature selection stage will refine features, which are input into a classification/learning process. Naïve Bayes and Support Vector Machines are usual supervised techniques in sentiment learning. Next section presents main concepts of these techniques as well as a short background on Artificial Neural Networks (ANN).

4. Classifiers

In this section we review fundamental aspects of three popular supervised classifiers: Naïve Bayes, Support Vector Machines and Artificial Neural Networks. Instead of providing a detailed description of these approaches, we focus on reviewing concepts of SVM and ANN with the purpose of discussing important issues in a fundamental comparative analysis (see Section 4.4). Despite the low computational cost of the Naïve Bayes technique, it has not been competitive in terms of classification accuracy when compared to SVM (Abbasi et al., 2008; Pang et al., 2002) and therefore we only analyze the NB method comparatively in the context of our experimental results. More details on models for text classification and NB classifiers can be found in McCallum and Nigam (1998), Manning et al. (2008).

4.1. Naïve Bayes

Naïve Bayes is a probabilistic learning method that assumes terms occur independently. Given a collection of N documents $\{d_j\}_{j=1}^{N}$, where each document is represented as a sequence of T terms $d_j = \{t_1, t_2, \ldots, t_T\}$, the probability of a document d_j occurring in class c_k is given as

$$P(c_k|d_j) = P(c_k) \prod_{i=1}^{T} P(t_i|c_k), \tag{3}$$

where $P(t_i|c_k)$ is the conditional probability of term t_i occurring in a document of class c_k and $P(c_k)$ is the prior probability of a document occurring in class c_k. $P(t_i|c_k)$ and $P(c_k)$ are estimated from the training data.

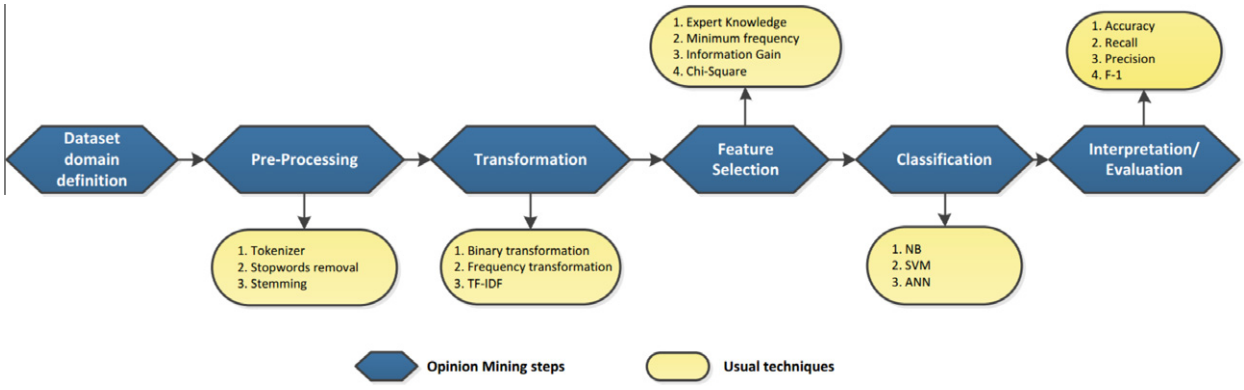

Fig. 1. Steps and techniques that are commonly found in sentiment classification approaches.

4.2. Support Vector Machines

Support Vector Machines is a supervised learning technique with many desirable qualities that make it a popular algorithm. It has a solid theoretical foundation and performs classification more accurately than most other algorithms in many applications. Many researchers have reported that SVM is perhaps the most accurate method for text classification (Liu, 2011). It is also widely used in sentiment classification (Tsytsarau & Palpanas, 2012).

SVM is a linear learning method that finds an optimal hyperplane to separate two classes. As a supervised classification approach, SVM seeks to maximize the distance to the closest training point from either class in order to achieve better generalization/classication performance on test data (Hastie, Tibshirani, & Friedman, 2001). The solution is based only on those training data points which are at the margin of the decision boundary. These points are called support vectors and are illustrated in Fig. 2(a). Instead of minimizing a global error function in a gradient descent process, which suffers from the existence of multiple local minima solutions, the parameters of the optimal separating hyperplane can be obtained by solving a convex optimization problem, for which there are standard software packages available.

When classes cannot be linearly separated, as shown in Fig. 2(b), the input data space is transformed into a higher-dimensional feature space in order to make data linearly separable and suitable for the linear SVM formulation. Usually, this transformation is achieved by using a *kernel function h* (Huang, Kecman, & Kopriva, 2006). It makes possible to determine a nonlinear decision boundary, which is linear in the higher-dimensional feature space, without computing the parameters of the optimal hyperplane in a feature space of possibly high dimensionality (Haykin, 1998). Therefore, the solution can be written as a weighted sum of the values of certain kernel function evaluated at the support vectors (Horváth, 2003).

4.3. Artificial Neural Networks

The central idea of a neural network is to derive features from linear combinations of the input data, and then model the output as a nonlinear function of these features (Hastie et al., 2001). The result is one of the most popular and effective forms of learning system (Russell, Norvig, & Davis, 2010).

Neural networks are typically represented by a network diagram which is composed of nodes connected by directed links. Nodes are arranged in layers and the structure of the most used neural network consists of three layers: an input, a hidden and an output layer of nodes (Hastie et al., 2001). It is also classified as a feed-forward network, since the nodes are connected only in one direction. Each connection has an associated weight, whose value is estimated by minimizing a global error function in a gradient descent training process (Haykin, 1998). Usually, a neuron is a simple mathematical model that produces an output value in two steps. First, the neuron computes a weighted sum of its inputs and then applies an *activation function* to this sum to derive its output (Russell et al., 2010). The activation function is typically a nonlinear function, and it ensures that the entire network can estimate a nonlinear function (e.g. a nonlinear decision boundary), which is learned from the input data.

4.4. Fundamental comparison between SVM and ANN

Formally, SVM and feed-forward neural networks are structurally similar, since both of them induce an output function which is expressed as a linear combination of simple functions (Romero & Toppo, 2007):

$$f(x) = b + \sum_{k=1}^{M} \lambda_k h(\omega_k, x). \tag{4}$$

The bias term b is common for both SVM and ANN (Haykin, 1998) and Table 1 explains the meaning of the remaining terms from both SVM and ANN points of view.

Despite structural similarities in the output function, the models differ in the way the solutions are obtained. The number M of support vectors is usually a result of the optimization problem posed, and the support vectors $\{\omega_k\}_{k=1}^{M}$ are always a subset of the data in the SVM algorithm. This property does not usually hold for ANN (Romero & Toppo, 2007) and the number M of hidden nodes in a neural network is a free parameter which is usually fixed previously. Therefore, in contrast to neural networks, SVM algorithm has a desired property of automatically selecting their model size M (by selecting the support vectors as a fraction of the training data). However, it is important to note that a large set of support vectors can be needed to form the output function, making SVM computationally slow in running time (test phase) and expensive for real-time applications (Burges, 1998; Romero & Toppo, 2007). Although the definition of an appropriate number M of hidden nodes in designing of a neural network is not a trivial task, the model complexity is usually controlled by keeping the number of hidden nodes small (Haykin, 1998). This is one of the issues under investigation in our work, i.e. *how is the performance of an ANN in comparison with a SVM model when the number of hidden neurons is a fraction of the number of support vectors in a sentiment classification task?*

An important advantage of SVM over ANN lies in the optimization approaches. SVM obtain the support vectors in a convex optimization problem, which always finds a global minimum and a unique solution, whereas ANN are trained with gradient descent

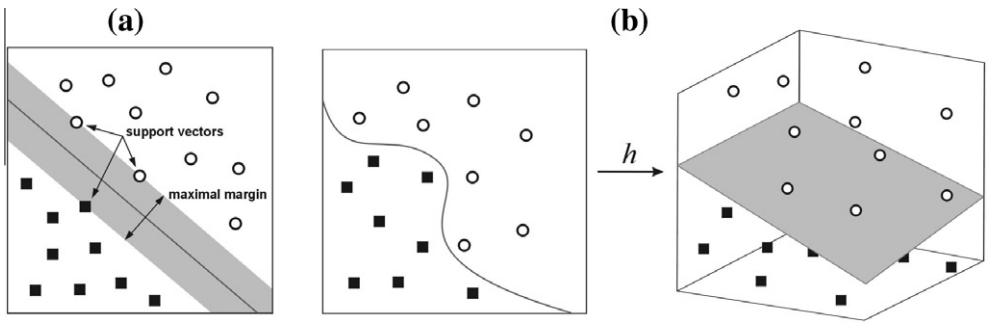

Fig. 2. Geometric principle of the SVM algorithm. (a) Linear SVM in a separable classification problem. Shaded region delineates the maximum margin which separates two classes based on three support points/vectors. (b) Nonlinear SVM. Transformation h from the original data space to the higher dimensional space so that a linear decision boundary can separate the two classes in the transformed space.

Table 1
Correspondence of terms in the output function induced by SVM and ANN (a single hidden layer network with linear output nodes) as expressed in Eq. (4).

Elements of Eq. (4)	From the SVM point of view	From the ANN point of view
M	Number of support vectors	Number of nodes in the hidden layer
h	Kernel function	Activation function
$\{\omega_k\}_{k=1}^{M}$	Support vectors	Hidden layer weights
$\{\lambda_k\}_{k=1}^{M}$	Coefficients found by the convex optimization problem	Output layer weights

methods, which may not converge to the optimal/global solution (Hastie et al., 2001; Haykin, 1998). However, some techniques have shown advances in minimizing the chance of local convergence. Although the scaled conjugate gradient method (Fodslette & Mnller, 1993) focus on accelerating the process of convergence, the reported results showed that the method failed to converge less often than traditional conjugate gradient or back-propagation using gradient descent.

5. Experiments

In this section, we report our experiments. We aim to not only compare best classification accuracies but also discuss contexts in which the classifiers produce comparable results, since in practice, models with good accuracy and low computational cost are desired. Section 5.1 describes the datasets used to train and test the models. The metrics used to evaluate the classification performance are described in Section 5.2. Section 5.3 presents our experiments set-up and parameter setting.

5.1. Datasets

We conducted experiments on four datasets, which are the benchmark Movies review dataset (Pang & Lee, 2004) and collections of reviews extracted from *amazon.com* in distinct product domains: GPS, Books and Cameras. Each dataset consists of 2000 reviews that were classified in terms of the overall orientation as being either positive or negative (1000 positive and 1000 negative reviews). The ground truth was obtained according to the customer 5-stars rating. Reviews with more than 3 stars were defined as being positive and reviews with less than 3 stars were labeled as being negative. Reviews with 3 stars are not included in our datasets. Table 2 characterizes the distribution of terms in the datasets after removing stopwords and stemming.

5.2. Performance measurement

We evaluate the classification performance in terms of three commonly used metrics: accuracy, recall and precision as defined in Eqs. (5)–(7) and Table 3. Table 3 is a confusion matrix whose entries are given as a function of two typical classes in document-level sentiment classification, positive and negative documents.

Table 2
Details of the datasets used in our experiments.

Domain	Number of distinct terms	Average number of terms per document
Movies	25456	665.6
GPS	6880	171.5
Books	10422	189.9
Cameras	5996	122.6

Table 3
Confusion matrix.

	Predicted	
	Positive documents	Negative documents
Actual positive documents	# True Positive samples (TP)	# False Negative samples (FN)
Actual negative documents	# False Positive samples (FP)	# True Negative samples (TN)

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{5}$$

$$\text{recall} = \frac{TP}{TP + FN}, \tag{6}$$

$$\text{precision} = \frac{TP}{TP + FP}. \tag{7}$$

5.3. Design of experiments

We focus on comparing SVM and ANN learning models when they achieve the best classification accuracy. However, both models require a critical parameter setting, which is still a research issue. A detailed discussion covering parameter values and their implications are beyond the scope of our work and can be found in Ben-Hur and Weston (2010), Burges (1998), Hastie et al. (2001), Haykin (1998). For each set of selected features, we perform an exhaustive searching through a subset of critical parameters values. Specifically, the parameter c in the SVM algorithm and the number of neurons M in the hidden layer of a neural network are empirically fitted in a grid search fashion guided by better values of accuracy.

We train the SVM classifier with an usual nonlinear kernel (radial basis function) using LIBSVM software package (Chang & Lin, 2011). We used default parameter values, except for the cost constant c, whose value is selected from the interval $c \in [10^{-1}, 10^{3}]$.

For the ANN classifier, we used the traditional feed-forward network with a single hidden layer. The number of neurons in the hidden layer M is selected from the set $M \in \{15, \ldots, 55\}$. In order to overcome the problem of convergence to a satisfactory solution, our exhaustive procedure of selecting parameters involved training candidate models three times, each of which started with a different set of randomly generated weights. Repeated training with random starting weights is a popular method to overcome the convergence problem (Atakulreka & Sutivong, 2007). We used the back-propagation algorithm to train the neural network models. Instead of adopting the traditional gradient descent method, we use the scaled conjugated gradient to speed up the convergence to a solution (Fodslette & Mnller, 1993), as implemented in the Matlab software. In order to accelerate the training process and reduce the risk of overfitting, we adopted the early stopping procedure (Bishop, 2007).

Following most of the experimental reports in the literature, our results are obtained in terms of 10-fold cross-validation. The preprocessing of the datasets consisted of removing stopwords and stemming by applying the Snowball *stemmer* (Porter, 2001), resulting in the vocabulary sizes described in Table 2. We used single words as features and TF-IDF as the weighting approach. For each set of training data, we refine features by using the information gain ranking (IG), as discussed in Section 3, and then evaluate the performance of the learning methods as a function of different sets of selected features.

6. Empirical results

We conduct experiments on balanced and unbalanced data. Our results are reported as the average of the test folds. Each of them

consist of 100 positive and 100 negative reviews. The remaining 1800 reviews (900 positive and 900 negative) are reserved for training the classifiers. Most of our results are given as a function of vocabulary sizes, since we aim to compare the behavior of classifiers, and their requirements to achieve better levels of accuracy, as a function of the number of input terms/dimensions. The vocabularies consist of terms that were best ranked by the IG technique in the training stage. We arbitrarily chose seven quantity of terms between 50 and 5000. In order to evaluate how different is the accuracy between SVM and ANN classifiers, we applied the *t* student test with 5% of significance (Alpaydin, 2004).

6.1. Results on balanced data

In this section, we report results by considering an equal number of positive and negative reviews in the training stage of the learning algorithms. Tables 4–7 show results for Movies, GPS, Books and Cameras datasets as a function of vocabulary sizes (i.e. number of terms). Figs. 3 and 4 illustrate the average classification accuracy behavior for each classifier.

Fig. 5 shows the average training and running time in seconds, as a function of different number of selected terms. In order to facilitate the visual comparison between ANN and SVM, we did not include the NB classifier, since it is much faster than ANN and SVM, as shown in Tables 4–7. Since we conducted the validation as a 10-fold process on 2000 reviews, training time is computed by considering 1800 reviews and running time is computed on the remaining 200 reviews. As discussed in Section 4.4, the computational cost of the SVM and ANN algorithms is connected with the number M of support vectors and neurons in the hidden layer respectively. Table 8 shows the requirements of SVM and ANN in terms of these variables to achieved the accuracy values reported in our experiments.

Figs. 6 and 7 summarizes the performance of ANN and SVM in terms of recall and precision. Figs. 6 and 7 show results as a function of the number of selected terms that resulted in the best accuracy for ANN and SVM respectively. Considering our results on balanced data, we observed the following:

1. An IG-based selection of more than 1000 terms did not result in a significant improvement in classification accuracy. Figs. 3–5 indicate that a number of terms ranging from 500 to 1000 can be a reasonable trade-off between computational effort and good levels of accuracy.
2. Although the accuracy difference between ANN and SVM has never exceeded 3%, ANN outperformed SVM significantly (t-test with $p < 0.05$) in 13 out of 28 tests, while SVM outperformed significantly ANN in only 2 tests, as shown in Figs. 3 and 4. In addition, ANN has achieved the best accuracy in all datasets, as shown in Tables 4–7.
3. As expected, the training time of ANN increased much faster than the training time of SVM when measured as a function of the number of terms. As the number of dimensions (i.e. input terms) increases, the training time of ANN scales up exponentially, as shown in Fig. 5(a).
4. However, Fig. 5(b) indicates that the running time of SVM increased faster than the running time of ANN when measured as a function of the number of terms. It may be a consequence of the high number of support vectors extracted from the training data, as shown in Table 8 and discussed in Section 7.
5. Although Naïve Bayes classifier produced the best absolute values of recall and precision, Tables 4–7 indicate that such behavior results from a good performance on a single class. In this context, SVM and ANN produced better results, since they were less prone to producing a biased classification.
6. Figs. 6 and 7 indicates that ANN and SVM produced comparable results regarding recall and precision. It should be noted that good values of recall and precision must be so high as similar for both classes, since it indicates an accurate and unbiased classification. Although the results vary from contexts in which ANN and SVM produce quite similar values (like in the Cameras dataset) to contexts in which ANN slightly outperforms SVM and vice versa (like in the Books dataset), an overall evaluation of Figs. 6 and 7 shows that ANN tend to be less biased than SVM.

6.2. Results on unbalanced data

In this section, we present results produced by unbalanced training processes. We focus on analyzing the behavior of SVM and ANN regarding different ratios of positive and negative reviews. In practice, it is reasonable to opt for a learning method that produce satisfactory results for both balanced and unbalanced data contexts. Therefore, we report results on unbalanced data by considering only the numbers of selected terms that resulted in the best accuracy on balanced data. Note, however, it is not a guarantee of good results regarding unbalanced data, since the feature selection will run on unbalanced data and therefore it is reasonable assume that the sets of selected terms will

Table 4
Movies dataset: Average results of a 10-fold cross validation as a function of the number of selected terms. POS and NEG are the positive and negative classes of reviews respectively. Best results are in boldface.

Classifiers	Number of terms						
	50	100	500	1000	3000	4000	5000
Accuracy							
ANN	80%	82.5%	86%	86%	**86.5%**	85.6%	85.8%
SVM	78.8%	82.6%	84.1%	85.2%	83.7%	83.7%	84.1%
NB	78.2%	80.3%	76%	72.5%	71.5%	71.3%	71.3%
Training time							
ANN	2.3 s	3.2 s	6.7 s	12.9 s	40.3 s	51.5 s	65.5 s
SVM	0.27 s	0.63 s	1.24 s	2.2 s	4.3 s	4.75 s	5.6 s
NB	**0.02 s**	**0.02 s**	0.05 s	0.07 s	0.15 s	0.18 s	0.23 s
Recall (POS:NEG)							
ANN	0.81:0.79	0.84:0.81	0.85:**0.87**	0.85:0.86	0.86:**0.87**	0.85:0.86	0.86:0.85
SVM	0.82:0.76	0.83:0.82	0.85:0.83	0.87:0.84	0.83:0.84	0.83:0.85	0.83:0.85
NB	0.75:0.81	0.86:0.75	0.96:0.56	0.97:0.48	**0.98**:0.45	**0.98**:0.45	**0.98**:0.45
Precision (POS:NEG)							
ANN	0.79:0.81	0.82:0.83	**0.87**:0.85	0.86:0.86	**0.87**:0.86	0.86:0.85	0.85:0.86
SVM	0.77:0.81	0.82:0.83	0.83:0.85	0.84:0.86	0.84:0.83	0.84:0.83	0.85:0.83
NB	0.8:0.77	0.78:0.85	0.69:0.94	0.65:0.94	0.64:**0.95**	0.64:**0.95**	0.64:**0.95**

Table 5
GPS dataset: Average results of a 10-fold cross validation as a function of the number of selected terms. POS and NEG are the positive and negative classes of reviews respectively. Best results are in boldface.

Classifiers	Number of terms						
	50	100	500	1000	3000	4000	5000
Accuracy							
ANN	80.1%	83.6%	86.5%	**87.3%**	85.7%	85.2%	85.2%
SVM	79.8%	83.2%	84.7%	84.5%	84.3%	83.9%	83.7%
NB	68.8%	65.8%	65.2%	65.1%	65.1%	65.1%	65.1%
Training time							
ANN	3.1 s	4.6 s	11.1 s	14.5 s	45.9 s	64.5 s	75.4 s
SVM	0.2 s	0.5 s	0.6 s	0.8 s	1.1 s	1.2 s	1.3 s
NB	**0.02 s**	**0.02 s**	0.04 s	0.06 s	0.12 s	0.16 s	0.19 s
Recall (POS:NEG)							
ANN	0.85:0.75	0.85:0.82	**0.89**:0.84	0.87:0.87	0.87:0.84	0.87:0.83	0.88:0.82
SVM	0.81:0.78	0.84:0.83	0.86:0.83	0.87:0.82	**0.89**:0.8	**0.89**:0.79	**0.89**:0.78
NB	0.42:0.95	0.34:0.97	0.32:**0.99**	0.31:**0.99**	0.31:**0.99**	0.31:**0.99**	0.31:**0.99**
Precision (POS:NEG)							
ANN	0.77:0.83	0.83:0.85	0.85:**0.88**	0.87:0.87	0.85:0.87	0.84:0.87	0.83:**0.88**
SVM	0.79:0.81	0.83:0.84	0.84:0.86	0.83:0.86	0.82:**0.88**	0.81:**0.88**	0.81:**0.88**
NB	0.9:0.62	0.93:0.6	**0.96**:0.59	**0.96**:0.59	**0.96**:0.59	**0.96**:0.59	**0.96**:0.59

Table 6
Books dataset: Average results of a 10-fold cross validation as a function of the number of selected terms. POS and NEG are the positive and negative classes of reviews respectively. Best results are in boldface.

Classifiers	Number of terms						
	50	100	500	1000	3000	4000	5000
Accuracy							
ANN	75.6%	78.9%	80.5%	**81.8%**	80.8%	79.6%	79.2%
SVM	73.9%	78.4%	79.8%	80.9%	81.4%	81.7%	81.4%
NB	71.7%	74.4%	75.8%	76.2%	75.9%	76.1%	76.1%
Training time							
ANN	3.7 s	5.5 s	12.1 s	21.3 s	42 s	63.9 s	69.4 s
SVM	0.22 s	0.39 s	0.54 s	0.78 s	1.2 s	1.4 s	1.5 s
NB	**0.01 s**	0.02 s	0.03 s	0.05 s	0.13 s	0.15 s	0.2 s
Recall (POS:NEG)							
ANN	0.84:0.67	0.84:0.74	0.83:0.78	0.84:**0.8**	0.83:0.78	0.85:0.75	0.81:0.78
SVM	0.84:0.64	0.83:0.73	0.84:0.75	0.87:0.75	**0.88**:0.75	0.87:0.76	0.87:0.76
NB	0.71:0.72	0.74:0.75	0.78:0.74	0.78:0.74	0.78:0.74	0.78:0.74	0.78:0.74
Precision (POS:NEG)							
ANN	0.72:0.81	0.76:0.82	0.79:0.82	**0.81**:0.83	0.79:0.82	0.77:0.83	0.79:0.8
SVM	0.7:0.8	0.76:0.81	0.77:0.83	0.78:0.85	0.78:**0.86**	0.79:**0.86**	0.78:0.85
NB	0.72:0.72	0.75:0.75	0.75:0.77	0.75:0.78	0.75:0.77	0.75:0.77	0.75:0.77

Table 7
Cameras dataset: Average results of a 10-fold cross validation as a function of the number of selected terms. POS and NEG are the positive and negative classes of reviews respectively. Best results are in boldface.

Classifiers	Number of terms						
	50	100	500	1000	3000	4000	5000
Accuracy							
ANN	84.9%	86.5%	89.9%	**90.3%**	89.8%	89.6%	88.8%
SVM	85.1%	88.0%	88.8%	89.6%	89.8%	89.7%	89.9%
NB	81.1%	80.1%	81.9%	81.8%	81.8%	81.8%	81.8%
Training time							
ANN	4.5 s	5.6 s	11 s	18.8 s	45.2 s	54.6 s	77.2 s
SVM	0.2 s	0.3 s	0.4 s	0.6 s	0.9 s	1 s	1.1 s
NB	**0.02 s**	**0.02 s**	0.04 s	0.05 s	0.1 s	0.2 s	0.2 s
Recall (POS:NEG)							
ANN	0.87:0.82	0.89:0.84	0.9:0.89	0.91:0.89	0.91:0.88	**0.92**:0.87	0.9:0.87
SVM	0.88:0.82	0.89:0.87	0.9:0.88	0.91:0.88	**0.92**:0.88	**0.92**:0.87	**0.92**:0.88
NB	0.71:0.91	0.65:0.95	0.68:**0.96**	0.68:**0.96**	0.68:**0.96**	0.68:**0.96**	0.68:**0.96**
Precision (POS:NEG)							
ANN	0.83:0.87	0.85:0.89	0.9:0.9	0.9:0.91	0.89:0.91	0.88:**0.92**	0.8:0.9
SVM	0.83:0.87	0.87:0.89	0.88:0.9	0.89:0.91	0.88:0.91	0.88:0.91	0.88:0.91
NB	0.89:0.76	0.93:0.73	**0.94**:0.75	**0.94**:0.75	**0.94**:0.75	**0.94**:0.75	**0.94**:0.75

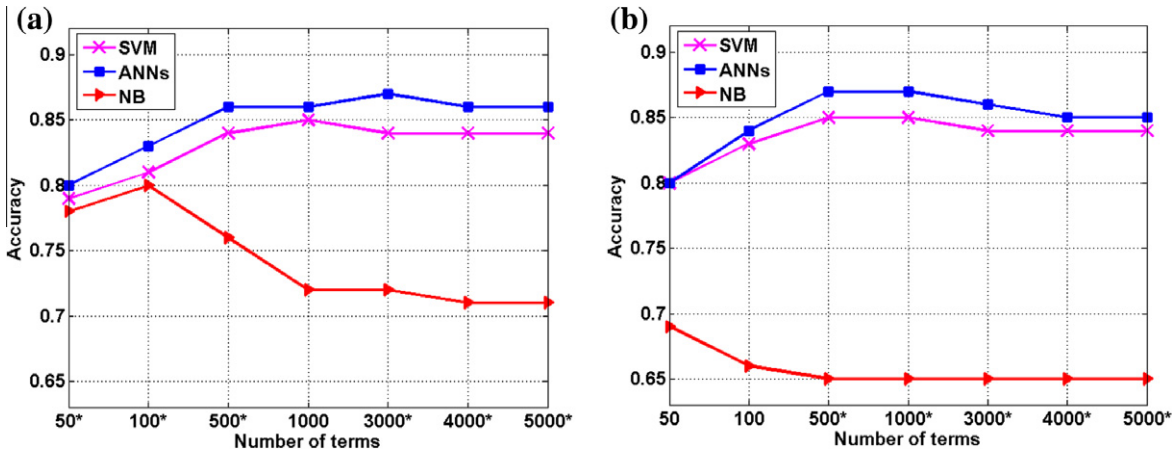

Fig. 3. Average classification accuracy as a function of the number of selected terms. '∗' indicates a significant difference between SVM and ANN (t-test with $p < 0.05$). (a) Movies dataset (b) GPS dataset.

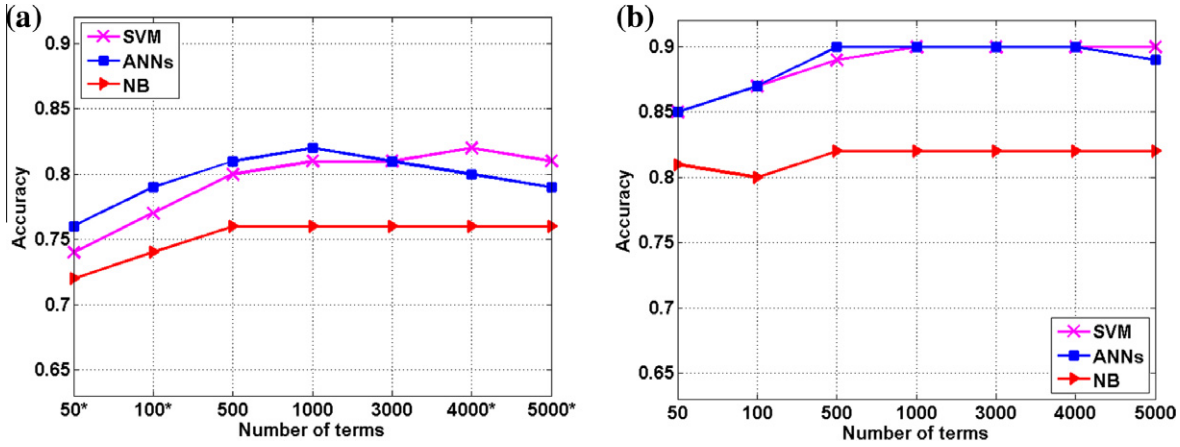

Fig. 4. Average classification accuracy as a function of the number of selected terms. '∗' indicates a significant difference between SVM and ANN (t-test with $p < 0.05$). (a) Books dataset (b) Cameras dataset.

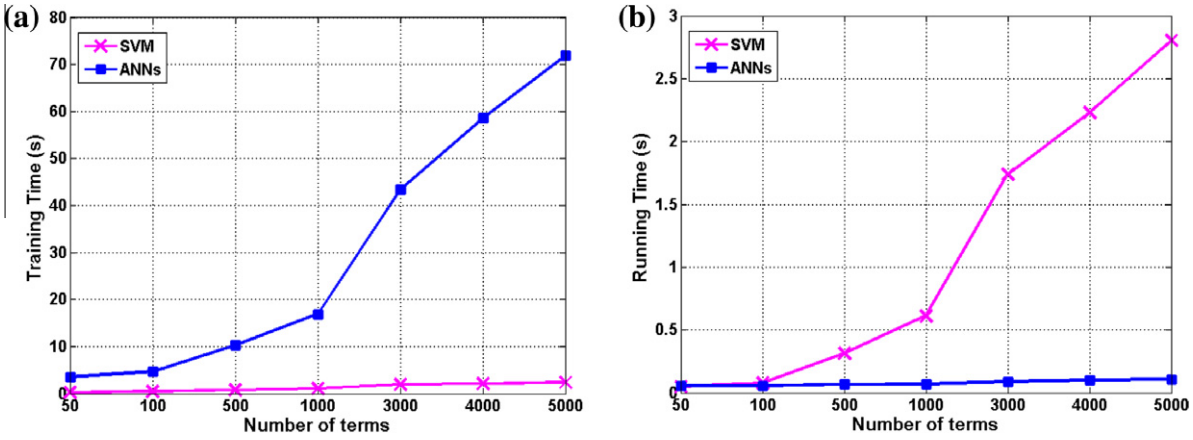

Fig. 5. Average training (a) and running (b) time as a function of the number of selected terms.

differ from those produced in the balanced data context. In order to approach realistic conditions of unbalanced data, we vary only the number of negative reviews in a training dataset, since in practice the number of positive reviews is substantially greater than the number of negative reviews (Burns et al., 2011; Glorot, Bordes, & Bengio, 2011).

Figs. 8–11 show the classification accuracy as a function of the proportion of negative reviews. Each figure shows two charts, which correspond to two selections of terms. The first selection resulted in the best classification accuracy on balanced data for ANN and the second selection produced the best classification accuracy on balanced data for SVM.

Table 8
Average number of support vectors and neurons in the hidden layer of the neural network.

Number of Terms	Number of support vectors	Number of hidden layer neurons
50	1097.1	30.5
100	1075.7	31.8
500	1051.3	31.8
1000	1143.1	29.4
3000	1210.0	24.7
4000	1220.8	24.2
5000	1233.1	25.6

Figs. 13 and 14 summarize the performance of ANN and SVM in terms of recall and precision for a data imbalance ratio of 80%, i.e. #NEG/ #POS = 0.2. Figs. 13 and 14 show results as a function of the number of selected terms that resulted in the best accuracy on balanced data for ANN and SVM respectively.

Considering our results on unbalanced data, we observed the following:

1. As expected, the classification accuracy of both methods decreased in proportion to the imbalance between the number of negative and positive reviews in the training phase (see Figs. 8–11).

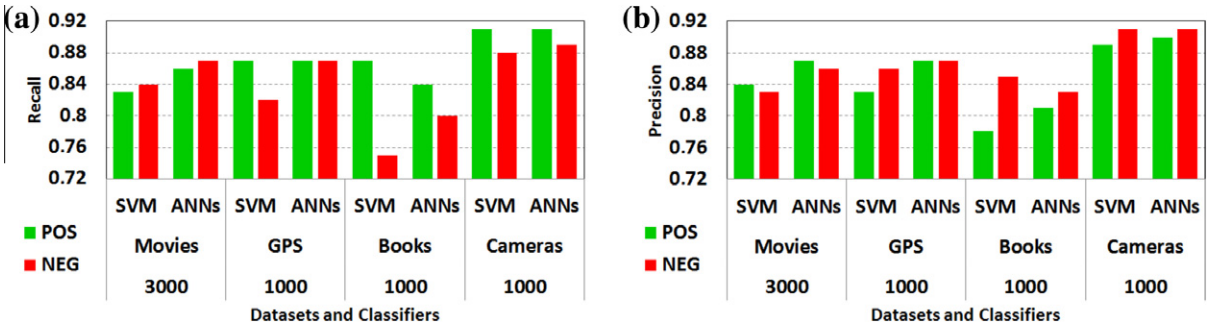

Fig. 6. Performance of ANN and SVM regarding (a) recall and (b) precision as a function of the number of selected terms that resulted in the best accuracy for ANN on balanced data.

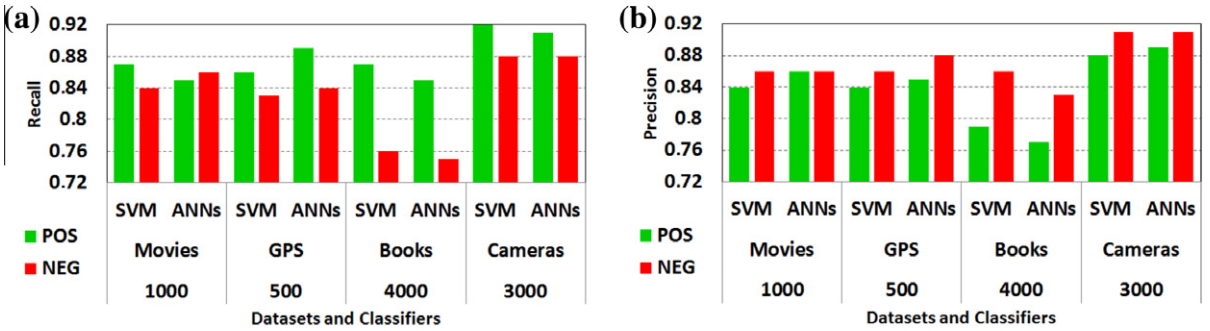

Fig. 7. Performance of ANN and SVM regarding (a) recall and (b) precision as a function of the number of selected terms that resulted in the best accuracy for SVM on balanced data.

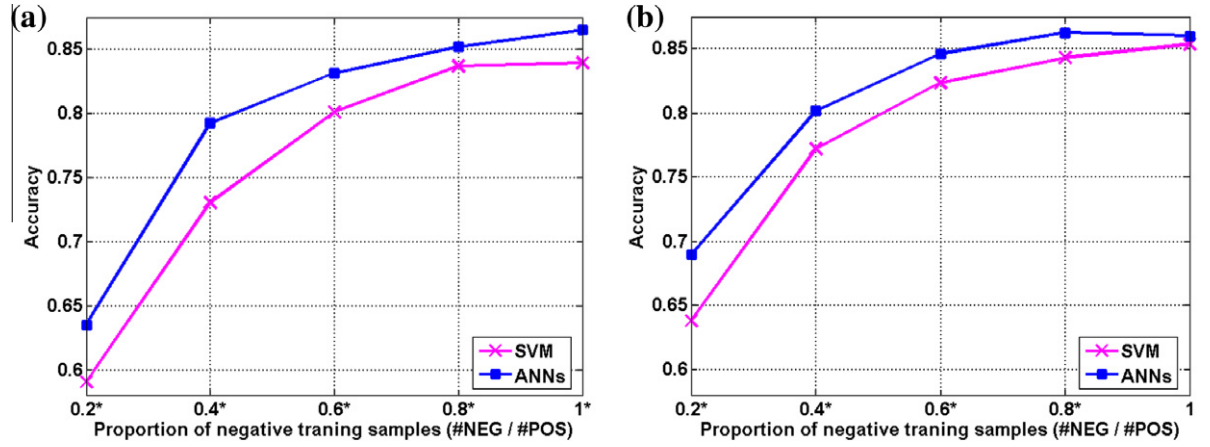

Fig. 8. Movies dataset: Average classification accuracy as a function of the proportion of negative reviews. (a) Accuracy produced from an IG-based selection of 3000 terms (b) Accuracy produced from an IG-based selection of 1000 terms.

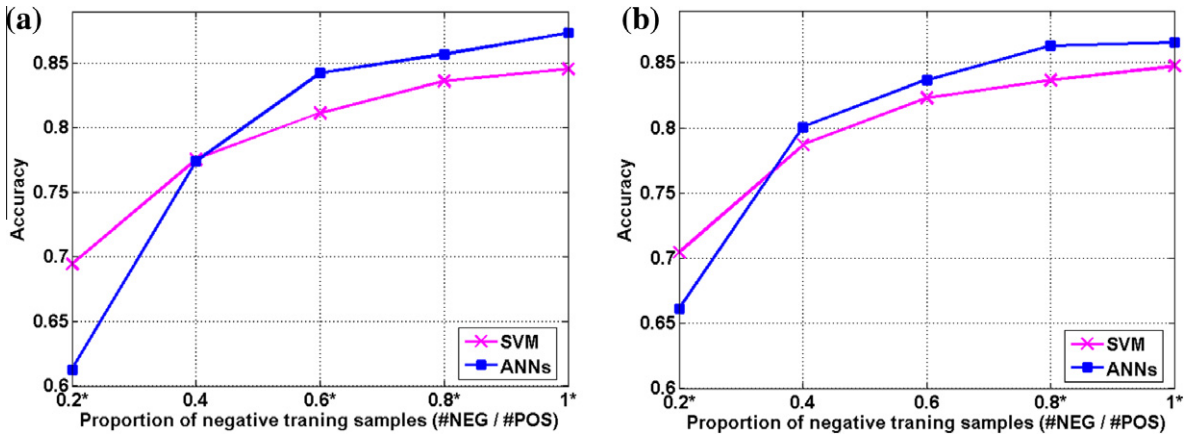

Fig. 9. GPS dataset: Average classification accuracy as a function of the proportion of negative reviews. (a) Accuracy produced from an IG-based selection of 1000 terms. (b) Accuracy produced from an IG-based selection of 500 terms.

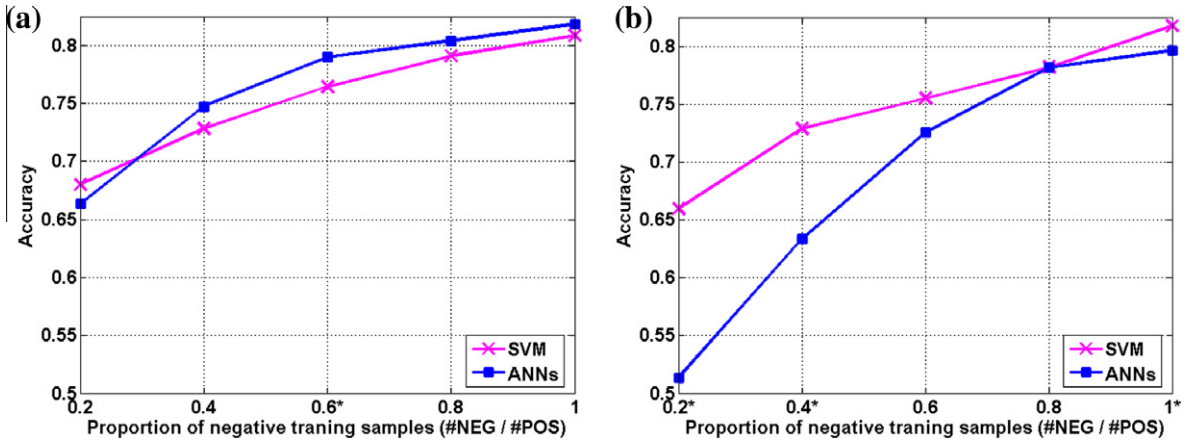

Fig. 10. Books dataset: Average classification accuracy as a function of the proportion of negative reviews. (a) Accuracy produced from an IG-based selection of 1000 terms. (b) Accuracy produced from an IG-based selection of 4000 terms.

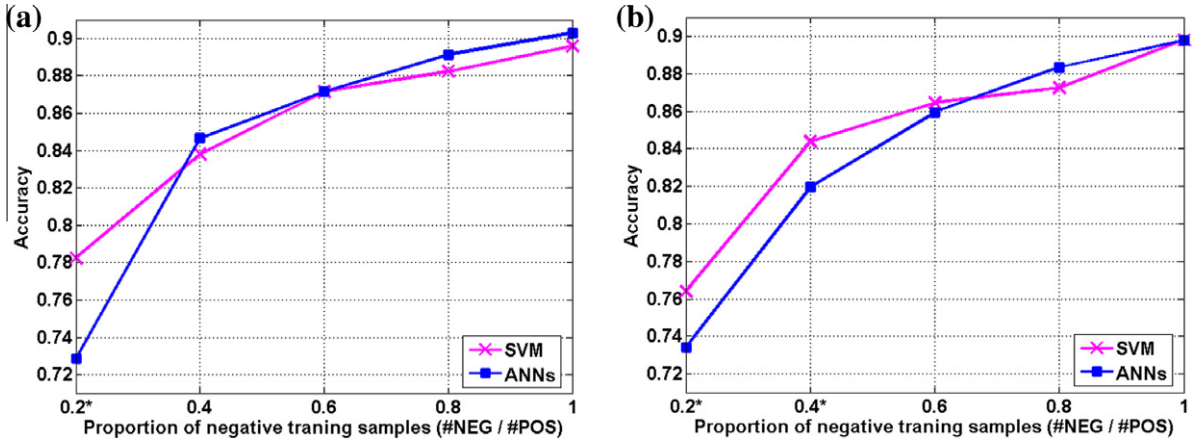

Fig. 11. Cameras dataset: Average classification accuracy as a function of the proportion of negative reviews. (a) Accuracy produced from an IG-based selection of 1000 terms. (b) Accuracy produced from an IG-based selection of 3000 terms.

2. ANN are more sensitive to unbalanced and noisy data than SVM. Since a selection of terms consist of the top ranked terms according to IG score (see Section 3), it is reasonable assume that the larger is a set of selected terms, the higher is the chance of it containing less important (noisy) terms. Figs. 9 and 11 indicate that the classification accuracy of ANN deteriorates as the level of data imbalance and the number of noisy terms increase (note the number of terms is shown in the caption of figures), while SVM tend to be less sensitive to these variables than ANNs. This may not be a reasonable conclusion in the context of Fig. 8, since ANN significantly outperformed SVM in all levels of data imbalance. However, the reason for this may be due to

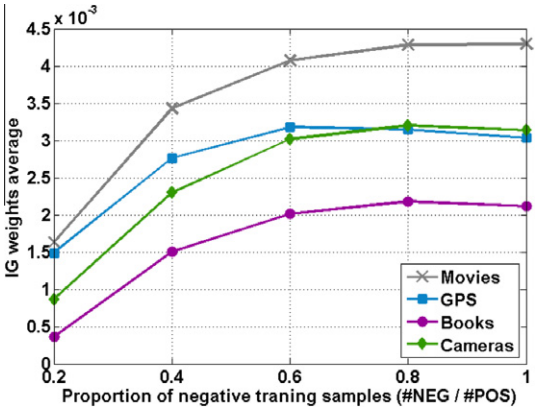

Fig. 12. Average of IG scores for the first 1000 terms as a function of data imbalance.

the quality of terms in the Movies dataset, since the reviews present characteristics that can result in a selection of terms with less noisy terms, e.g.: (i) reviews contain a high number of terms (see Table 2) and (ii) the terms reached the best average of IG scores when compared to the other datasets (see Fig. 12).

3. Since the training data are highly unbalanced, the results naturally tend to favor the majority class of positive reviews. Therefore, we focus on analyzing the recall of negative reviews, which measures the performance of classifiers in losing negative reviews, and the precision of positive reviews, which measures the performance of classifiers in misclassifying reviews as being of the majority class (i.e. how biased the classifier is towards the majority class). In general, our results indicate that SVM outperforms ANN, except in the Movies dataset. The reason may be due to the quality of terms, as discussed above.

7. Discussion of results

Although we have reported results only with an usual nonlinear kernel, our experiments indicate that SVM requires a high number of support vectors to classify reviews as expressing positive or negative opinions at document level (see Table 8). Therefore, it is reasonable to conclude that the classes cannot be well (and linearly) separated on the basis of single terms as the dimensions of the input space, resulting in a SVM's running time much higher than that of an ANN. Thus, although Cristianini and Shawe-Taylor (2000) mentioned that in practice SVM frequently results in very few support vectors, our results are consistent with previous results on text categorization (Colas, Pacĺık, Kok, & Brazdil, 2007) in reporting a high number of support vectors.

Neural networks has been rarely used as an approach for sentiment learning and one of the reasons for this can be the high computational cost of training them on high-dimensional data. However, our experiments indicate that a standard feature selection method (IG) performs well in the task of refining data to be input into a neural network and consequently reducing the computational effort in the training process. Our results on balanced data indicate that as the number of selected terms rise beyond 1000 terms, ANN not only present a quick increase of training time (see Fig. 5(a)) but also result in no significant improvements on the classification accuracy (see Figs. 3 and 4), indicating that the IG method perform satisfactorily in filtering noisy terms. On text categorization, some authors (Gabrilovich & Markovitch, 2004; Taira & Haruno, 1999) have recommended a SVM training process with all available features. However our experiments shown that SVM have also benefited from the IG-based term selection, since the running time of SVM can be significantly reduced by considering less input terms (see Fig. 5(b)). According to Figs. 3 and 4, there seems to be no need to involve more than a fraction of the vocabulary (less than 10%), since no significant improvements on the classification accuracy are achieved

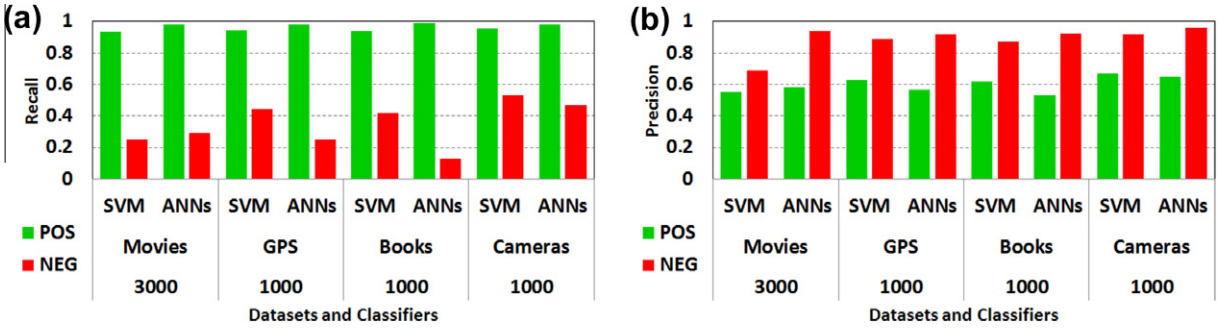

Fig. 13. Performance of ANN and SVM in terms of (a) recall and (b) precision for a training data imbalance ratio of #NEG/ #POS = 0.2. The horizontal axis consists of the number of selected terms that resulted in the best accuracy of ANN on balanced data.

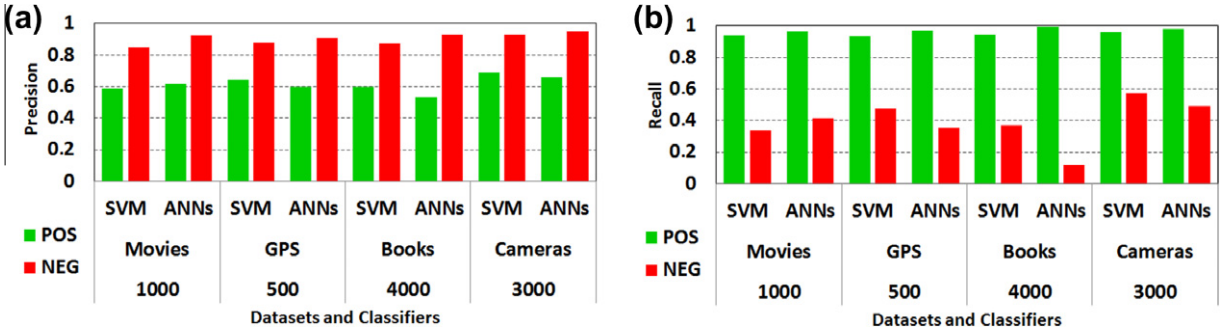

Fig. 14. Performance of ANN and SVM in terms of (a) recall (b) precision for a training data imbalance ratio of #NEG/ #POS = 0.2. The horizontal axis consists of the number of selected terms that resulted in the best accuracy of SVM on balanced data.

when considering an IG-based term selection of more than 500–1000 terms. In summary, although reducing input features is critical to make the ANN training process practical, it does not represent a disadvantage in comparison to SVM, since SVM also benefits from the feature selection (Abbasi et al., 2011, 2008; Dang et al., 2010; Li et al., 2009), especially in the context of large-scale sentiment classification tasks (Bespalov et al., 2011).

In order to evaluate ANN in terms of the convergence to a satisfactory solution, we adopted the strategy of training a neural network more than once. Our results come from an ANN model that result in the best accuracy among three competing models, which consist of the same number of nodes in the hidden layer but were trained with different starting weights. In practice, the requirement of training a neural network more than once is a disadvantage in comparison with SVM, since the SVM's optimization method always converge to a unique solution. Nevertheless, each ANN training process could run simultaneously in parallel as proposed in Atakulreka and Sutivong (2007). In addition, if the running time is an issue instead of the training time, a common scenario in practice, to use an ANN model to classify new data can be a good choice, since the ANN classification process (i.e. an ANN model at running time) can be much faster than the SVM classification process (see Fig. 5(b)).

Despite the issues discussed above, our results indicated that SVM tend to be more stable than ANN to deal with noisy terms in an unbalanced data context. Since it is reasonable to assume that Books, GPS and Cameras datasets have produced more noisy terms than the benchmark dataset of Movies reviews, the behavior of the classification accuracy as a function of an increasing data imbalance on these datasets shown that the performance of ANN tend to decrease below the performance of SVM, specially when the reviews are represented with more (noisy) terms (see Figs. 9 and 11).

8. Conclusion and future work

Support Vector Machine (SVM) has been widely and successfully used in sentiment analysis as neural networks (ANNs) has attracted little attention as an approach for sentiment learning. Literature has been reporting comparisons between SVM and ANN in various research areas, since there seems to be no clear consensus on an absolute winner method. Instead, there seems to be a connection between the best method and the underlying learning problem. To the best of our knowledge, a comparative study of ANN and popular sentiment learning approaches under the same framework is still missing. Therefore, we have presented an empirical comparison between a neural network approach and a SVM based-method for classifying positive versus negative-oriented reviews.

We have focused on comparing SVM and ANN in terms of the requirements to achieve better classification accuracies. Our experiments evaluated both methods as a function of selected terms in a bag-of-words (unigrams) approach. Regarding the sentiment learning literature, our main findings/contributions are in the following points:

- In terms of classification accuracy on the benchmark dataset of Movies reviews (Pang & Lee, 2004), ANN outperformed significantly (statistically) SVM, specially in the context of unbalanced data.
- As an overall comparison in the context of balanced data, we have performed a total of 28 tests on four datasets. ANN outperformed SVM significantly (t-test with $p < 0.05$) in 13 tests, while SVM outperformed significantly ANN in only 2 tests. Although the accuracy difference between them has never exceeded 3%, ANN has achieved the best classification accuracy in all datasets.
- However, our results indicated that SVM tend to be less affected by noisy terms than ANN when the data imbalance increases.
- As expected, the training time of a neural network is usually much higher than that of SVM. However, if the point at issue is the running time, ANN could be recommended, since SVM have resulted in a high number of support vectors and a consequent running time that grows much faster than the running time of ANN.
- Information gain, a computationally cheap feature selection method, can be used to reduce the computational effort of both ANN and SVM without affecting significantly the resulting classification accuracy. Considering an increasing number of input terms, our results have indicated the existence of thresholds above which little improvements in the resulting accuracy are achieved. Our results indicated that IG (i) makes the ANN training practical in a bag-of-words approach and (ii) contribute to reduce the running time of SVM, although the complexity of the SVM solution does not necessarily depend on the number of features (Joachims, 1998; Suykens, Vandewalle, & Moor, 2001).

In summary, our results indicate that ANN can be a candidate approach when the task involves sentiment learning. Future work will concentrate on three aspects. First, a comparative study between SVM and ANN by involving features like part-of-speech tags and joint topic-sentiment measurements. Second, the maximum entropy (ME) classification method (He et al., 2011) has shown promising results in sentiment analysis and therefore we intend to involve ME in our comparative study. Finally, our results on unbalanced data suggest a relation between data quality and the fact of ANN outperforming SVM. Therefore, we intend to investigate the potential of a feature selection method, like IG, in predicting the best classifier to be used as a function of data quality.

Acknowledgment

We thank to CNPq (Conselho Nacional de Desenvolvimento Cientifico e Tecnológico, Brazil) and CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) for financial support.

References

Abbasi, A. (2010). Intelligent feature selection for opinion classification. *IEEE Intelligent Systems, 25*, 75–79.

Abbasi, A., Chen, H., & Salem, A. (2008). Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Transactions on Information Systems, 26*, 12:1–12:34.

Abbasi, A., France, S., Zhang, Z., & Chen, H. (2011). Selecting attributes for sentiment classification using feature relation networks. *IEEE Transactions on Knowledge and Data Engineering, 23*, 447–462.

Alpaydin, E. (2004). *Introduction to machine learning*. Cambridge, Mass: Mit Press.

Atakulreka, A., & Sutivong, D. (2007). Avoiding local minima in feedforward neural networks by simultaneous learning. In *Proceedings of the Australian joint conference on advances in artificial intelligence* (pp. 100–109).

Baccianella, A. E. S., & Sebastiani, F. (2010). Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the 7th conference on international language resources and evaluation (LREC'10)*. Valletta, Malta: European Language Resources Association (ELRA).

Bai, X. (2011). Predicting consumer sentiments from online text. *Decision Support Systems, 50*, 732–742.

Ben-Hur, A., & Weston, J. (2010). A user's guide to support vector machines data mining techniques for the life sciences. *Methods in molecular biology (Clifton, N.J.), 609*, 223–239.

Berry, M. W., & Kogan, J. (Eds.). (2010). *Text mining: Applications and theory*. Chichester, UK: Wiley.

Bespalov, D., Bai, B., Qi, Y., & Shokoufandeh, A. (2011). Sentiment classification based on supervised latent n-gram analysis. In *Proceedings of the 20th ACM international conference on information and knowledge management* (pp. 375–382).

Bishop, C. M. (2007). *Pattern recognition and machine learning* (2nd ed.). Springer.

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery, 2*, 121–167.

Burns, N., Bi, Y., Wang, H., & Anderson, T. (2011). Sentiment analysis of customer reviews: Balanced versus unbalanced datasets. In *Knowledge-based and intelligent information and engineering systems. Lecture notes in computer science* (Vol. 6881, pp. 161–170). Berlin/ Heidelberg: Springer.

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology, 2*, 27:1–27:27.

Chen, L.-S., Liu, C.-H., & Chiu, H.-J. (2011). A neural network based approach for sentiment classification in the blogosphere. *Journal of Informetrics, 5*, 313–322.

Claster, W., Hung, D.Q., & Shanmuganathan, S. (2010). Unsupervised artificial neural nets for modeling movie sentiment. In *International conference on computational intelligence, communication systems and networks* (pp. 349–354).

Colas, F., Pacĺık, P., Kok, J. N., & Brazdil, P. (2007). Does SVM really scale up to large bag of words feature spaces? In *Proceedings of the 7th international conference on Intelligent data analysis* (pp. 296–307). Berlin, Heidelberg: Springer-Verlag.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines and other Kernel-based learning methods* (1st ed.). Cambridge University Press.

Dang, Y., Zhang, Y., & Chen, H. (2010). A lexicon-enhanced method for sentiment classification: An experiment on online product reviews. *IEEE Intelligent Systems, 25*, 46–53.

Fodslette, M., & Mnller (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks, 6*, 525–533.

Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research, 3*, 1289–1305.

Gabrilovich, E., & Markovitch, S. (2004). Text categorization with many redundant features: Using aggressive feature selection to make svms competitive with c4.5. In *Proceedings of the 21st international conference on machine learning* (pp. 321–328). Banff, Alberta, Canada: Morgan Kaufman.

Glorot, X., Bordes, A., & Bengio, Y. (2011). Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the international conference on machine learning* (pp. 513–520).

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The Elements of Statistical Learning*. New York, NY, USA: Springer New York Inc.

Hatzivassiloglou, V., & McKeown, K. (1997). Predicting the semantic orientation of adjectives (pp. 174–181).

Haykin, S. (1998). *Neural networks: A comprehensive foundation* (2nd ed.). Upper Saddle River, NJ: Prentice Hall.

He, Y., Lin, C., & Alani, H. (2011). Automatically extracting polarity-bearing topics for cross-domain sentiment classification. In *Proceedings of annual meeting of the association for computational linguistics* (pp. 123–131).

Horváth, G. (2003). Neural networks in measurement systems (an engineering view). In J. Suykens, G. Horváth, S. Basu, C. Micchelli, & J. Vandewalle (Eds.), *Advances in learning theory: Methods, models and applications*. *NATO science series. Series III: Computer and systems sciences* (Vol. 190, pp. 375–402). Amsterdam: IOS Press. chap. 18.

Huang, T. M., Kecman, V., & Kopriva, I. (2006). *Kernel based algorithms for mining huge data sets: Supervised, semi-supervised, and unsupervised learning. Studies in computational intelligence* (Vol. 17). Secaucus, NJ, USA: Springer.

Joachims, T. (1998). Text categorization with suport vector machines: Learning with many relevant features. In *Proceedings of the 10th European conference on machine learning* (pp. 137–142).

Li, S., Wang, Z., Zhou, G., & Lee, S.Y.M. (2011a). Semi-supervised learning for imbalanced sentiment classification. In *Proceedings of international joint conference on artificial intelligence* (pp. 1826–1831).

Li, S., Xia, R., Zong, C., & Huang, C.-R. (2009). A framework of feature selection methods for text categorization. In *Proceedings of the 47th annual meeting of the ACL* (pp. 692–700).

Li, S., Zhou, G., Wang, Z., Lee, S.Y.M., & Wang, R. (2011b). Imbalanced sentiment classification. In *Proceedings of ACM international conference on information and knowledge management* (pp. 2469–2472).

Lin, C., & He, Y. (2009). Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM conference on information and knowledge management* (pp. 375–384).

Liu, B. (2011). *Web data mining: Exploring hyperlinks* (2nd ed.). New York: Springer.

Lovins, J. B. (1968). Development of a stemming algorithm. *Mechanical Translation and Computational Linguistic*, 22–31.

Manning, C. D., Raghavan, P., & Schtze, H. (2008). *Introduction to information retrieval*. New York, NY, USA: Cambridge University Press.

McCallum, A., & Nigam, K. (1998). A comparison of event models for naive Bayes text classification. In *Proceedings of the AAAI-98 workshop on learning for text categorization* (pp. 41–48).

O'Keefe, T., & Koprinska, I. (2009). Feature selection and weighting methods in sentiment analysis. In *Proceedings of the Australasian document computing symposium* (pp. 67–74).

Paltoglou, G., & Thelwall, M. (2010). *A study of information retrieval weighting schemes for sentiment analysis* (pp. 1386–1395).

Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the association for computational linguistics* (pp. 271–278).

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

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 conference on empirical methods in natural language processing* (pp. 79–86).

Porter, M. (1980). An algorithm for suffix stripping. *Program, 14*, 130–137.

Porter, M. (2001). Snowball: A language for stemming algorithms.

Prabowo, R., & Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics, 3*, 143–157.

Raychev, V., & Nakov, P. (2009). Language-independent sentiment analysis using subjectivity and positional information. In *Proceedings of the international conference RANLP-2009* (pp. 360–364).

Riloff, E., Patwardhan, S., & Wiebe, J. (2006). Feature subsumption for opinion analysis. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 440–448).

Romero, E., & Toppo, D. (2007). Comparing support vector machines and feedforward neural networks with similar hidden-layer weights. *IEEE Transactions on Neural Networks, 18*, 959–963.

Russell, S. J., Norvig, P., & Davis, E. (2010). *Artificial intelligence: A modern approach*. Prentice Hall.

Suykens, J., Vandewalle, J., & Moor, B. D. (2001). Optimal control by least squares support vector machines. *Neural Networks, 14*, 23–35.

Taira, H., & Haruno, M. (1999). Feature selection in svm text categorization. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 480–486).

Tang, H., Tan, S., & Cheng, X. (2009). A survey on sentiment detection of reviews. *Expert Systems with Application, 36*, 10760–10773.

Tsytsarau, M., & Palpanas, T. (2012). Survey on mining subjective data on the web. *Data Mining and Knowledge Discovery, 24*, 478–514.

Turney, P. D. (2002). In *Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. Proceedings of the 40th annual meeting on association for computational linguistics ACL '02* (pp. 417–424). Stroudsburg, PA, USA: Association for Computational Linguistics.

Van Hulse, J., Khoshgoftaar, T.M., & Napolitano, A. (2007). Experimental perspectives on learning from imbalanced data. In *Proceedings of the international conference on machine learning* (pp. 935–942).

Wang, Z., Li, S., Zhou, G., Li, P., & Zhu, Q. (2011). Imbalanced sentiment classification with multi strategy ensemble learning. In *Proceedings of international conference on Asian language processing* (pp. 131–134).

Weiss, S. M., Indurkhya, N., & Zhang, T. (2004). *Text mining. Predictive methods for analyzing unstructured information* (1st ed.,). Berlin: Springer.

Weiss, S. M., Indurkhya, N., & Zhang, T. (2010). *Fundamentals of predictive text mining*. London; New York: Springer-Verlag.

Whitelaw, C., Garg, N., & Argamon, S. (2005). Using appraisal groups for sentiment analysis. In *Proceedings of the 14th ACM international conference on information and knowledge management* (pp. 625–631).

Xia, R., & Zong, C. (2010). Exploring the use of word relation features for sentiment classification. In *Proceedings of the international conference on computational linguistics* (pp. 1336–1344).

Xia, R., Zong, C., & Li, S. (2011). Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences, 181*, 1138–1152.

Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In *Proceedings of the 14th international conference on machine learning ICML '97* (pp. 412–420). San Francisco, CA, USA: Morgan Kaufman.

Ye, Q., Zhang, Z., & Law, R. (2009). Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. *Expert Systems with Application, 36*, 6527–6535.

Zaidan, O.F., Eisner, J., & Piatko, C. (2007). Using "annotator rationales" to improve machine learning for text categorization. In *Proceedings of the association for computational linguistics* (pp. 260–267).

Zhu, J., XU, C., & shi Wang, H. (2010). Sentiment classification using the theory of anns. *The Journal of China Universities of Posts and Telecommunications, 17*, 58–62.