A Computational Intelligence Approach to Efficiently Predicting Review Ratings in E-Commerce

Georgina Cosma and Giovanni Acampora

School of Science and Technology, Nottingham Trent University, Nottingham, NG11 8NS, United Kingdom

Abstract

Sentiment Analysis, also called Opinion Mining, is currently one of the most studied research fields which aims to analyse people's opinions. E-commerce websites allow users to share opinions about a product/service by providing textual reviews along with numerical ratings. These opinions greatly influence future consumer purchasing decisions. This paper introduces an innovative computational intelligence framework for efficiently predicting customer review ratings by addressing two important issues involved in this significant task: the dimension and imprecision of customer textual review data. In particular, the proposed framework integrates the techniques of Singular Value Decomposition (SVD) and dimensionality reduction, Fuzzy C-Means (FCM) and the Adaptive Neuro-Fuzzy Inference System (ANFIS). The performance of the proposed approach returned high accuracy and the results revealed that when large datasets are concerned, only a fraction of the data is needed for creating a system to predict the review ratings of textual reviews. Results from the experiments suggest that the proposed synergetic approach yields better prediction performance than other state-of-the-art rating predictors which are based on the conventional Artificial Neural Network, Fuzzy C-Means, and Support Vector Machine approaches. In addition, the proposed framework can be utilised for other classification and prediction tasks, and its neuro-fuzzy predictor module can be replaced by other classifiers.

Keywords: Customer review ratings prediction, data mining, imprecision in customer reviews, fuzzy approach, machine learning, computational intelligence

1. Introduction

E-commerce companies reach out and gain customers via their electronic commerce websites. Customers consider information about products and services, available on company websites, for making informed decisions about purchases. Customers prefer to gather information online than from in-store due to the richness of the information available to them over the Internet [1], [2], [3], [4], [5], [6]. To facilitate this preference, e-commerce websites and online product review websites allow users to share opinions about a product/service by providing textual reviews along with numerical ratings. These opinions greatly influence future consumer purchasing decisions, because many consumers take into consideration reviews as a reliable resource when deciding whether to buy a product.

Online product reviews are also a potentially valuable source of information for companies, since companies use this information to monitor customer attitudes toward their products/services. Based on this information, companies can adapt their manufacturing, distribution, and marketing strategies accordingly. For these reasons, e-commerce companies consider reviews and review ratings as important and influential information to potential buyers, and thus encourage users to provide considerate and accurate reviews. In order to encourage reviewers to provide useful and informative reviews, some companies (e.g. epinion.com) reward those reviewers who provide useful reviews by giving them a status and/or financial rewards. These approaches are adopted by companies to reduce the occurrence of incorrect/inconsistent data recorded about products, since this data can affect the derived statistics about a product.

It is important that textual reviews match their corresponding numerical ratings in order to have a consistent system. By automatically predicting the numerical rating of each textual review, the accuracy of the data recorded can be improved. The fast growing number of online product review forums has attracted research into approaches for mining these new sources of information for decision support. Machine learning approaches (ML), and supervised learning approaches in particular, have been applied to review rating prediction [7], [8], [9] and opinion mining classification and such techniques can achieve a level of accuracy comparable to that achieved by human experts [10]. Most recent predictors are mainly based on Artificial Neural Network (ANN) approaches, and are not capable of dealing with the dimensioanlity and imprecision which is apparent in textual review data.

This paper introduces an innovative computational intelligence framework that comprises of an integration of different intelligent methodologies, able to efficiently reduce the size of textual review datasets and to analyse the 'imprecise' human sentiments hidden in the reviews. In particular, the proposed framework uses Singular Value Decomposition (SVD) and Dimensionality Reduction to extract the important and semantic features from each review and to consequently reduce the size and complexity of the entire reviews dataset. Then the Fuzzy C-Means (FCM) clustering algorithm classifies the refined reviews into fuzzy clusters in order to create an initial collection of rating prediction rules. Finally, this preliminary set of rules are used to start the Adaptive Neuro-Fuzzy Inference System (ANFIS) learning stage, which creates an optimised reviews rating prediction system.

The paper is organised as follows: Section 2 describes the most relevant literature, Section 3 describes the proposed Computational Intelligence Predictor framework, Section 4 discusses the experiment methodology. Section 5 discusses the experiment results and compares the performance of the proposed system with other approaches. A conclusion and outline of future work is provided in Section 6.

2. Literature Review

Sentiment Analysis, also called Opinion Mining, is currently one of the most studied research fields which aims to analyse people's opinions and emotions on products, individuals, organizations, and services [11]. The task of providing numerical ratings to textual reviews using a multi-point rating scale is referred to as rating-inference, belonging to the research area of Opinion Mining. An overview of existing literature on the topic of Opinion Mining (or sentiment analysis) is presented in [7],[8],[9], and [12]. Most of the existing rating inference methods proposed are most successful for binary classification of reviews (i.e. positive or negative) and less research has been accomplished on the topic of classifying reviews on a multi-point rating scale. Machine learning (ML) approaches, and in particular supervised learning approaches, have been applied to predict customer review ratings.

ML approaches which have been applied to opinion mining classification can achieve a level of accuracy comparable to that achieved by human experts [10]. Gamon [13] proposed a system for automatic sentiment classification of noisy customer feedback data. Their system achieved good accuracy when using large feature vectors in combination with feature reduction and SVM-based classification. More specifically, for classification of documents as belonging to rating category 1 versus rating category 4 a 85.47% accuracy was reported. However, for classification of documents belonging to categories 1 or 2 versus 3 or 4 a 69.23% accuracy was reported. Prabowo and Thelwall [9] proposed a method combining the rule-based classification and supervised learning approaches. They performed a comparative study using multiple classifiers, and examined the benefits and drawbacks of ML-based classification approaches. Their results revealed that hybrid approaches based on multiple classifiers can improve classification performance. Ye et. al. [14] compared three supervised ML algorithms (Naive Bayes, Support Vector Machine (SVM), and character-based N-gram model) for sentiment classification of online reviews on travel blogs and found that the SVM and n-gram approaches outperformed the Naive Bayes approach. Moraes [15] compared ANNs with SVMs for the task of classifying reviews as positive or negative. Their results revealed that ANNs outperformed the SVMs, and raised the limitations of these techniques, which were the computational cost of SVM at running time and that of ANN at the training time. Other researchers have also applied SVM for the ratings classification task [16], [17]. Saggion et.al. [17] have applied a number of text summarisation approaches within the rating-inference task on a corpus containing bank reviews. The reviews were rated on a scale of 1 to 5 by real users. The SVM learning algorithm was used to predict the correct rating of the full reviews and the reviews of automatically produced summaries. Although preliminary, their findings suggested that query-focused and sentiment-based summaries may be suitable for tackling the rating inference problem. Benamara et.al. [18], proposed an *adverb-adjective combinations* (AAC) sentiment analysis technique which uses a linguistic analysis of adverbs of degree (such as extremely, absolutely, hardly, precisely, really) that affect the intensity of adjectives. They proposed a methodology for scoring adverbs by defining a set of general axioms based on a classification of adverbs of degree into five categories. Similarly, Pang and Lee [19], proposed a technique that is based on metric labelling, which alters a given classifier's output in order to give similar labels to similar items. Leung et.al. [20] proposed framework for extracting the orientations and strength of opinion words. Their approach is based on part-of-speech tagging, negation tagging, feature generalisation and an opinion dictionary that uses a relative frequency-based method. Concerning the rating inference task, a score is calculated by assigning weights to different opinion words according to their estimated importance. They evaluated their proposed framework in rating reviews using 2-point and 3-point scales. The authors claim that the proposed framework appears to be effective for determining overall sentiments of products reviews. However, as noted by Pang and Lee [8] there is a noticeable difference between rating inference and predicting strength of opinion, which is essentially the aim of the framework proposed by [20] – "for instance you can feel quite strongly (high on the strength scale) that something is mediocre (middling on the "evaluation" scale)" [8]. More recently, Ochi [21] proposed a method of improve the prediction accuracy on the rating prediction task by correcting the bias of user ratings. The bias of the rating is detected using entropy of user rating and by updating word weights only when the words appear in the review, the problem of bias is reduced. Ganu [22] proposed methods for deriving a text-based rating from reviews and clustered similar users together using the topics and sentiments that appear in the reviews. In particular, they utilised soft clustering techniques to cluster 'like-minded' users for personalised recommendations, using the textual structure and sentiment of reviews.

3. A Computational Intelligence Predictor for Customer Review Ratings in E-Markets

This section introduces an innovative computational intelligence framework for predicting customer review ratings in e-commerce scenarios. A dataset of customer reviews could contain imprecise information due to intrinsic human vagueness and errors. The proposed framework illustrated in Figure 1, integrates techniques to efficiently tackle these issues. At the *Learning phase*, the *Natural Language Processing (NLP) module* prepares the data, the *Input Selection module* removes noise from the data to model the customers' reviews in a much reduced dimensional space, and the *Neuro-Fuzzy module* is efficiently applied to a reduced dimensional space to classify the reviews. Thereafter, a detailed description of each module is provided, and demonstrated via a small example. At the *Prediction Phase*, the framework takes new reviews and predicts their numerical rating.

3.1. Natural Language Processing (NLP) Module

A significant number of reviews are provided by consumers via popular e-commerce portals, such asAmazon.com or e-Bay.com. These reviews are accessed and used on a daily basis by millions of people. The storage size and processing power required to analyse these reviews is immense. For this

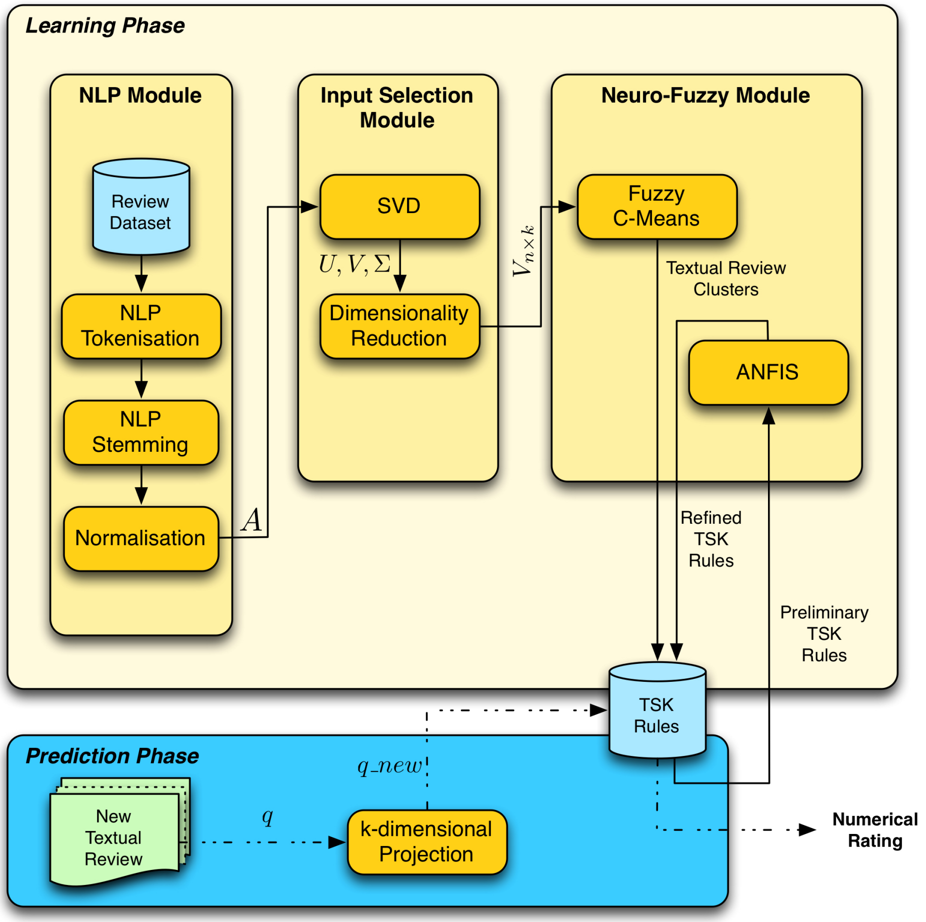

Figure 1: Computational Intelligence System Architecture for Rating Prediction

reason, the Natural Language Processing Module aims to reduce the dataset size in order to decrease the computational complexity of mining the data. The first step towards reducing the size of data involves the application of Natural Language Processing techniques. Initially, all upper-case letters are converted to lower case, and tokenisation and stemming [23] are applied to the collection of textual reviews contained in an e-commerce portal dataset. In particular, for each textual review, *Tokenisation* involves separating joint terms into multiple terms, for example, the term *niceradio* becomes two terms *nice* and *radio*. *Stemming* involves transforming variants of terms with the same root into the same term, and the Porter's stemming algorithm was adopted. In addition, terms that are solely composed of numeric characters, syntactical tokens (i.e. semicolons, and colons) and punctuation marks, terms that occur in only one review (global threshold), and terms with length equal to one are all removed. For example, consider the following reviews.

- R1: A great little product. Nice sleek design, love it.
- R2: I absolutely love this product, it is perfect. Nice colour.
- R3: A great product, easy to use, nice colour and sleek design.
- R4: It is annoying that I could not get it to charge. Unfortunately I have had to return it since it did not work.
- R5: There is a white line right in the middle which is annoying. Colour is nice.
- R6: Very disappointed the product was faulty and would not charge.
- R7: What a waste of money!
- R8: The button did not work. I had to return it such a waste of money.

Applying stemming changes the dictionary size. This is because the frequency of terms may increase if they are transformed into their stemmed format. Take for example two reviews – one contains the term *find*, and the other contains the word *finding*, and assume that the frequency occurrence of each of these terms across the entire review collection is 1. If no stemming is applied both of these terms will be removed from the dictionary as an initial mechanism for reducing dictionary size. However, if stemming is applied, both terms will be transformed into their stemmed format, which is the term *find*, and the local frequency of this term in each review will be 1, and its global frequency will be 2. After stemming is applied, the term will not be removed from the dictionary since its frequency is greater than 1. Applying tokenisation, stemming, and term-weighting, resulted in 36 dictionary terms, with an average number of 7.375 indexing terms per review.

Once a collection of refined terms have been extracted form the original textual review, the NLP Module uses a *Vector Space Model* to index the information. The information is presented as a term-by-review matrix $A_{m\times n} = [a_{ij}]$, in which each row i holds the frequency of refined term in textual reviews, and each column j represents a textual review. Hence, each cell a_{ij} of A contains the frequency at which a term i appears in a review j. The next goal is to normalise the term frequency in matrix A. In particular, a *global weighting* function is used to adjust the frequency of textual review terms in respect to the entire collection of reviews. At the same way, *review length normalisation* is applied to tune the frequencies based on the length of each review. In detail, this module uses the *normal global weighting* function named g_i and the *cosine document length normalisation* named n_j:

$$g_i = \frac{1}{\sqrt{\sum_j a_{ij}^2}}$$
$$n_j = (\textstyle\sum_i (g_i \cdot a_{ij})^2)^{-1/2}$$

where $a_{ij} = A[i, j]$. After the normalisation step is performed, each entry of the matrix A is updated as follows:

$$a_{ij} = a_{ij} \times g_i \times n_j$$

with $i = 1, \ldots, m$ and $j = 1, \ldots, n$. The role of normalisation is crucial because it enables the framework to capture information about the importance of each term in describing each textual review. In particular, if $a[i_1, j] \geq a[i_2, j]$ then the term i_1 is more significant in describing the j^{th} textual review than the term related to review i_2.

3.2. Input Selection Module

After the Natural Language Processing (NLP) Module has performed an initial reduction of the dataset size and it has created the term-by-review matrix A, the Input Selection Module further reduces the space complexity by removing noise and irrelevant data. This task is accomplished by using the Singular Value Decomposition and the Dimensionality Reduction statistical techniques, both described in [24]. The joint exploitation of these techniques enables the Input Selection Module to further reduce the computational time for training the Neuro-Fuzzy module. In particular, SVD decomposes the normalised $m \times n$ matrix A into the product of three other matrices:

$$A_{m\times n} = U_{m\times r} \cdot \Sigma_{r\times r} \cdot V_{r\times n}$$

where U is an $m \times r$ term-by-dimension matrix, Σ is an $r \times r$ singular values matrix and V is an $n \times r$ reviews-by-dimension matrix and r is the rank of the matrix A.

The Input Selection Module completes its task by providing a rank-k approximation to matrix A, where k represents the number of dimensions (or factors) retained, and $k \leq r$. In order to achieve this goal, the Input Selection Module uses a process known as *dimensionality reduction*, which involves truncating all three matrices to k dimensions. The dimensionality reduction technique applies the Cattell's graphical Scree test [25] on the singular values contained in the Σ matrix in order to determine the optimal number of the value of k.

3.3. Neuro-Fuzzy Module

The Neuro-fuzzy module takes as input the reduced review-by-dimension matrix $V_{r \times k}$ and it is trained to predict the review ratings of textual reviews using a a neuro-fuzzy learning algorithm. The learning algorithm works in two sequential steps. In the first step, the Fuzzy c-Means (FCM) clustering algorithm is applied to generate a collection of textual review clusters where each cluster contains the review characterised by a similar collection of qualitative terms. By using the approach proposed by Sugeno and Yasukawa [26], a collection of Takagi-Sugeno-Kang (TSK) rules, one TSK rule for each cluster, are generated for determining the membership of a review to a particular cluster. The second step uses this collection of rules as input to the Adaptive Neuro-Fuzzy Inference System (ANFIS) algorithm. ANFIS opportunely tunes the fuzzy rules and the related fuzzy membership functions in order to generate an optimised predictor model for textual reviews.

ANFIS was proposed by Jang [27] and it is a fuzzy system belonging to the adaptive networks framework. The aim of the ANFIS model is to transform human knowledge into a rule based fuzzy inference system, and to address the need for effective methods for tuning membership functions in order to minimise the output error. It creates an input-output mapping based on fuzzy if-then rules and on input-output data pairs by using a hybrid gradient-descent and least squares algorithm. Once trained, ANFIS can be used to solve various problems including prediction.

Formally, let $V_{r \times k} = [v_1, v_2, v_3, \ldots, v_n]$ be the reviews-by-dimension matrix, and let $2 \leq c < n$ be an integer, where c is the number of clusters and n is the total number of reviews. The FCM algorithm returns a list of cluster centres $X = x_1, \ldots, x_c$ and a membership matrix $U = \mu_{i,k} \in [0, 1],\ i = 1, \ldots, n,\ k = 1, \ldots, c$, where each element μ_{ik} holds the total membership of a data point v_k belonging to cluster c_i. FCM updates the cluster centers and the membership grades for each data point, and iteratively moves the cluster centers to the wright location within a dataset. This iteration is based on minimizing an objective function that represents the distance from any given data point to a cluster center weighted by that data point's membership grade. The objective function for FCM is a generalisation of equation (1)

$$J(U, c_1, \ldots, c_c) = \sum_{i=1}^{c} \sum_{k=1}^{N} \mu_{ik}^{m} ||v_k - x_i||^2, 1 \leq m \leq \infty \tag{1}$$

where μ_{ik} represents the degree of membership of review v_i in the ith cluster; x_i is the cluster centre of fuzzy group i; $|| * ||$ is the Euclidean distance between ith cluster and jth data point; and $m \in [1, \infty]$ is a weighting exponent. The necessary conditions for function (1) to reach its minimum are shown in functions (2) and (3).

$$c_i = \frac{\sum_{k=1}^{N} \mu_{ik}^{m} v_k}{\sum_{k=1}^{N} \mu_{i,k}^{m}}, \tag{2}$$

$$\mu_{ik} = \frac{1}{\sum_{k=1}^{c} \left(\frac{||v_k - x_i||}{||v_k - x_i||} \right)^{2/(m-1)}}, \tag{3}$$

A Sugeno-type Fuzzy Inference System (FIS) is generated using FCM clustering. The number of clusters determines the number of rules and membership functions in the generated FIS. The FIS has a network-type structure which maps inputs through input membership functions and associated parameters, and then through output membership functions and associated parameters to outputs. Thus, the membership degree of a review determines how close a review is to the next cluster.

Let c_a and c_b be two clusters, a review v_k can belong to cluster c_a such that $v_k \in c_a$, or it can belong in the intersection area between two clusters such that $v_k \in c_a \wedge v_k \in c_b$. The output FIS is fed into the ANFIS and the FIS parameters are tuned using the input/output training data in order to optimise the model. ANFIS uses a hybrid learning algorithm to identify the membership function parameters of single-output Sugeno type FIS. The training process stops whenever the designated epoch number is reached or the training error goal is achieved. The performance of ANFIS is evaluated using the array of root mean square training errors (difference between the FIS output and the training data output) at each epoch. During the learning process, the parameters associated with the membership functions are tuned using a gradient vector which, given a set of parameters, measures the performance of the system by means of how well it models input and output data. The architecture of a Type-3 ANFIS is explained in [27].

Assume that for each of the reviews specified in the small example found in Section 3.1, there exists a corresponding review rating found in vector $R_{8\times 1}$ where $R \in [1, 2]$ and R contains the following ratings R=[2;2;2;1;1;1;1;1] (note that a rating value of 1 indicates a negative review and a value of 2 is a positive review). Due to the size of the example dataset and for demonstration purposes the number of classes has been set to 2. The input to the ANFIS was review-by-dimension matrix $V_{8\times 2}$ and the target outputs vector $R_{8\times 1}$.

In this small example, the predicted values, PV, returned by the system for each of the review are $PV = [2.07; 1.75; 2.04; 1.23; 1.29; 1.10; 1.17; 1.36]$. Notice that for the textual review "R5: There is a white line right in the middle which is annoying. Colour is nice." cannot be really classified as positive or negative and hence the value returned by the predictor system reflects this nicely, considering the size of the example.

The PV values can then be compared to the actual rating values using various evaluation measures to determine the accuracy of the system. These measures are described in Section 4.3. The results for the small dataset returned a very low error rate which is close to zero, $RMSE = 0.000252$. This is a very small dataset and a zero-error rate is much less likely for a large dataset reviews. In addition, it is worth noting that applying FCM only for the particular small dataset revealed a higher error rate ($RMSE = 0.3386$) indicating that applying ANFIS does have a positive and promising impact on prediction. In this paper, the Computational Intelligence Predictor will be evaluated on large datasets to determine its efficiency.

3.4. Prediction Phase

The Prediction phase takes a new review and predicts its numerical rating. Since the Input Selection Module reduces the dimensionality of the original term-by-review matrix A, when a new review is input into the system to be classified, it needs to be transformed into a term-by-review vector and projected to the reduced dimensional space V_k. Thus, given a review vector q, whose non-zero elements contain the normalised term frequency values of the terms, a new review vector q_{new} can be obtained from the projection of q to the k-dimensional space [24] as follows:

$$q_{new} = q^T \times U_k \times \Sigma_k^{-1} \tag{4}$$

Table 1: Dataset characteristics

Dataset Name	**Dictionary size**	**No. of reviews**	**No. of reviews selected for the experiments**
Wrist Watches	16,547	68,355	30,000
Jewellery	10,333	58,621	40,000
Software	11,210	95,084	80,000
Total	38,090	222,060	150,000

Once projected, a review is represented as a review vector q_{new} of size k. This means that SVD and dimensionality reduction need only be recomputed once the size of the reviews dataset stored in the dataset increases significantly.

4. Experiment Methodology

The section discusses the experiment methodology adopted to efficiently evaluate the performance of the proposed system against other approaches. In particular, Section 4.1 provides details of the datasets used for the experiments, Section 4.2 discusses how the K-fold cross validation approach has been applied to ensure efficient evaluation, Section 4.3 discusses the evaluation measures utilised to measure system performance, Section 4.4 discusses alternative computational intelligence classifiers which have been applied for review ratings prediction. The results of the experiments are presented in Section 5.

4.1. Datasets

The datasets used for experimentation consist of customer reviews and their corresponding numerical ratings. Each rating is mapped to a numerical rating on a scale ranging from $1-5$. The datasets[1] used for the experiments are publicly available and have been created by researchers for opinion mining and sentiment analysis tasks. All datasets contain the textual reviews and the ratings provided by customers for those reviews. The reason for using real datasets rather than artificial ones is to determine the accuracy of the classifiers in a real setting when minimal human pre-processing is carried out.

[1] Available at: https://snap.stanford.edu/data/

Table 4.1, contains the characteristics of the datasets which were utilised for the experiments. The column titles of each table are explained as follows: *Dataset Name* is the name of each dataset, *Dictionary size* is the total number of terms found in the entire dataset after the NLP module is applied. These terms are used to construct the dictionary which was utilised for training the system. *No. of reviews* is the total number of reviews found in the entire dataset. *No. of reviews selected for experimentation* is the total number of reviews selected from a dataset and used for conducting the experiments.

4.2. Evaluation Approach: K-fold Cross Validation

After applying Singular Value Decomposition and dimensionality reduction, the reviews-by-dimension matrix V of each dataset was partitioned into subdatasets of $10,000$ reviews. The reason for partitioning the matrix into subdatasets was to reduce the complexity of training the neuro-fuzzy predictor module using very large datasets, and to reduce the time required to train the classifier.

During the K-fold cross validation process, each subdataset is partitioned into 4 subsets, with each subset, V_i, having $2,500$ reviews. The validation process runs over K iterations, where during each iteration, subset V_i is reserved as the test set, and the remaining partitions are collectively used to train the model. Hence, in the first iteration the system is trained on subsets V_2, V_3, V_4 and tested on subset V_1; in the second iteration the system is trained on subsets V_1, V_3, V_4 and tested on V_2; and the process continues. The system's performance is evaluated using the classification results of the test subsets across all K iterations (i.e. folds), using the evaluation metrics described in Section 4.3.

4.3. Evaluation Measures

This section presents the evaluation measures adopted for assessing the review rating prediction performance of the proposed classifier and other classifiers. Table 2 holds the evaluation measure formulas and the following notation relates to the evaluation measures. Note that, all reviews which belong to a given rating class are called the positive reviews of that class, whereas all other reviews in the dataset are considered as negative for that class. For example, in review rating class 1, all reviews which have been rated as class 1 by human reviewers are the positives for that class, and all other reviews which do not belong to class 1 are negatives for that class.

- Let $|TP|$ be the total number of true positives retrieved by the classifier. These are the positive cases that were correctly labelled by the classifier.
- Let $|TN|$ be total the number of true negatives retrieved by the classifier. These are the negative cases that were correctly labelled by the classifier.
- Let $|FP|$ be the total number of false positives retrieved by the classifier. These are the negative cases that were incorrectly labelled by the classifier as positive.
- Let $|FN|$ be the total number of false negatives retrieved by the classifier. These are the positive cases that were incorrectly labelled by the classifier as negative.
- Let $|P|$ be the total number of positive cases that exist in the dataset, where $|P| = |TP| + |FN|$.
- Let $|N|$ be the total number of negative cases that exist in the dataset, where $|N| = |FP| + |TN|$.

Table 2: Performance evaluation measures

Evaluation Measure	**Formula**
Precision	$\frac{\lvert TP\rvert}{\lvert TP\rvert+\lvert FP\rvert}$
Recall, sensitivity, true positive rate	$\frac{\lvert TP\rvert}{\lvert P\rvert}$
Accuracy, recognition rate	$\frac{\lvert TP\rvert+\lvert TN\rvert}{\lvert P\rvert+\lvert N\rvert}$
Misclassification rate, error rate,	$\frac{\lvert FP\rvert+\lvert FN\rvert}{\lvert P\rvert+\lvert N\rvert}$ or $1 - Accuracy$
F_1-score, harmonic mean of precision and recall	$\frac{2\times Precision\times Recall}{\lvert P\rvert+\lvert N\rvert}$

Recall (R) is the fraction of positive reviews in the dataset which have been correctly classified as positive. An inefficient system can achieve high Recall. This is because the purpose of Recall is to determine how many of the positive cases have been correctly classified, ignoring the number of FP cases retrieved. Precision (P), on the other hand, is the fraction of reviews classified as positive which are true positive, hence it is a measure of exactness. These measures alone are not sufficient enough to determine the accuracy of the classifier and additional evaluation measures (see Table 2) must be adopted for an efficient evaluation.

The measure of Accuracy evaluates the performance of the system as the fraction of its classifications that are correct - it considers the number of true positives and true negatives over all classified cases. The highest the value of accuracy the better the performance of the classification system. *Percentage of misclassification* gives the percentage of reviews which have been misclassified. Misclassification is computed as 1-Accuracy, and the lower the value the better the performance of the classification system. In addition, the Root Mean Squared Error (RMSE) is another measure used for evaluating classification system accuracy. The RMSE measure estimates the residual between the actual and predicted values. A model has better performance if it has a smaller RMSE. An RMSE equal to zero represents a perfect fit. RMSE is computed as follows,

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(t_i - y_i)^2} \tag{5}$$

where t_i is the actual (desired) value, y_i is the predicted value produced by the model, and m is the total number of observations.

Finally, in order to appropriately consider all the values presented, a new error measure, E, is introduced. Measure E combines the misclassification error and the RMSE into a single measure. A better system would receive lower values for both of these measures. Let ϕ be the fraction of reviews that have been misclassified, and let ϵ be the root mean squared error, then the performance of a system is computed as function (6)

$$E = \frac{1}{2} \cdot \sum_{i=1}^{m} \phi + \epsilon \tag{6}$$

where E be the new error value, and m is the total number of observations.

4.4. Alternative Classifiers

In the experiments, the Neuro-Fuzzy module of the proposed framework, described in Section 3, was replaced by the Fuzzy-C means (FCM), Artificial Neural Network (ANN) and the Support Vector Machine Classifier (SVM), in order to compare the performance of the proposed system against alternative approaches. Sections 4.4.1 and 4.4.2 provide a brief description of the ANN and SVM classifiers, respectively, and explain how these have been tuned to achieve their best performance for the review ratings prediction task when using the proposed framework.

4.4.1. Artificial Neural Network

A *m-by-n* matrix representing the n sample reviews of m elements (i.e. each element is a term), and the target data consisting of a *1-by-n* vector where each element represents the value of a rating corresponding to an input vector, were used for training the feedforward Artificial Neural Network (ANN). Each input is weighted with an appropriate weight w. The sum of the weighted inputs plus the bias forms the input to the transfer function f. The tan-sigmoid transfer function was used at the hidden layer, and the linear transfer function was used at the output layer. Once the weights and biases of the network are initialised the network is ready for training. Details behind the mathematics of the feedforward ANN and the Scaled Conjugate Gradient (SCG) training algorithm can be found in [28] and [29] respectively. Regarding the ANN parameters used for the experiments, the tan-sigmoid transfer function was used at the hidden layer, and the linear transfer function was used at the output layer of the ANN model. The ANN was trained by using the SCG for Fast Supervised Learning suitable for large-scale problems [29]. With an ANN, over-fitting can occur when there are too many neurons in the hidden layer and for this reason, in order to avoid the risk of generalisation, experiments were conducted with various number of neurons to determine optimal number of neurons to use for achieving highest performance. Increasing the number of neurons impacts on processing time, and taking into consideration time and performance, 10 neurons were a good choice for all datasets.

4.4.2. Support Vector Machine

Support Vector Machines (SVM), is a method for the classification of both linear and nonlinear data. It uses nonlinear mapping to transform the original training data (pattern vectors) into a higher dimensional feature space. Within this new dimension it searches for the maximum marginal hyperplane which serves as the best boundary for separating the data into two classes. The best hyperplane for an SVM means the one with the largest margin between the two classes. The support vectors are the data points that are closest to the separating hyperplane. The one-against-all binary classifier has been applied for the review ratings prediction problem. The one-against-all approach builds one SVM per class, trained to distinguish the samples in a single class from the samples in all remaining classes. Hence multiple binary classifiers are combined to solve the multi-class classifier problem using SVM [30]. For many real-world practical problems there may be no linear boundary separating the classes and hence an optimal separation of hyperplanes may not be reached. Experiments have been conducted with different SVM kernel functions which are capable of performing linear and nonlinear hyperplane separation. These included the: Linear kernel, Quadratic kernel, Polynomial kernel (default order 3), Gaussian Radial Basis Function kernel (with scaling factors, sigma, of 1 and 16), and Multilayer Perceptron (MLP) kernel with scale [1 -1]. Experimental results returned higher model performance when the MLP kernel function was used to map the training data into kernel space, across all datasets. The MLP kernel function was the only one which could reach convergence for all datasets, whereas the rest of the kernel functions were able to reach convergence for some but not all datasets.

5. Experiment Results

This section discusses the results of experiments which were conducted to evaluate the performance of the proposed computational intelligence predictor against alternative models. In each experiment, the Neuro-Fuzzy module of the proposed framework (FCM+ANFIS), described in Section 3, was replaced by the Fuzzy C-Means (FCM), Artificial Neural Network (ANN) and the Support Vector Machine Classifier (SVM), in order to compare the performance of the proposed system against alternative classifiers. All of the alternative classifiers were tuned to achieve their best performance when using the proposed framework because to allow for a fair comparison among the different approaches. Table 3 shows the average performance of each classifier across each dataset. The results of applying each classifier to each dataset partition (i.e. subdataset) are shown in Tables 4, 5, and 6.

When comparing the results, emphasis was placed on the *% of misclassified*, *RMSE*, and the proposed evaluation measure *E* (which is the average of the misclassification error and RMSE). The RMSE is also considered to be an important measure for the proposed sentiment analysis application because the difference between the actual and predicted values determines how close each classifier is in predicting the value provided by the human user. The reason for placing more emphasis on the *E* and *RMSE* evaluation measures is because the differences among the classifiers FCM+ANFIS, FCM, ANN, when using all other evaluation measures, was very close.

On average, across all datasets, the proposed classifier FCM+ANFIS achieved better results when compared to all other classifiers - its performance was better on 12 out of 15 partitions (all partitions for the Wrist Watches dataset; 2 out of 4 partitions for the Jewellery dataset; and 7 out of 8 partitions for the Software dataset). The performance of every classifier depends on the dataset, and no one single classifier is suitable for all datasets. Despite this, FCM+ANFIS performed consistently better than all other classifiers, returning lower error values (i.e. *RMSE*, *E*).

Table 3: Average performance of each classifier across all datasets

Dataset 1: Wrist Watches								
Wrist Watches	Recall	Precision	Accuracy	F_1-measure	%Misclassified	RMSE	E	Rank E
FCM+ANFIS based	0.924	0.863	0.864	0.890	13.632	1.239	0.688	1
FCM based	0.916	0.857	0.859	0.882	14.150	1.277	0.709	3
ANN based	0.911	0.873	0.862	0.890	13.833	1.268	0.703	2
SVM based	0.642	0.812	0.617	0.705	38.289	2.455	1.419	4
Dataset 2: Jewellery								
Jewellery	Recall	Precision	Accuracy	F_1-measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.940	0.887	0.886	0.911	11.356	1.176	0.645	2
FCM based	0.895	0.876	0.872	0.877	12.830	1.233	0.680	3
ANN based	0.932	0.881	0.885	0.905	11.500	1.165	0.640	1
SVM based	0.929	0.808	0.764	0.845	23.619	1.816	1.026	4
Dataset 3: Software								
Software	Recall	Precision	Accuracy	F-measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.956	0.858	0.849	0.904	13.274	1.343	0.7380	1
FCM based	0.971	0.841	0.840	0.901	16.026	1.374	0.7673	3
ANN based	0.932	0.873	0.847	0.901	15.316	1.363	0.7582	2
SVM based	0.744	0.837	0.675	0.780	32.454	2.226	1.2755	4
Average across all datasets								
Average	**Recall**	**Precision**	**Accuracy**	F_1 **measure**	**%Misclassified**	**RMSE**	E	Rank E
FCM+ANFIS based	0.940	0.870	0.866	0.902	12.754	1.253	0.690	1
FCM based	0.928	0.858	0.857	0.887	14.335	1.295	0.719	3
ANN based	0.925	0.876	0.865	0.899	13.549	1.266	0.700	2
SVM based	0.772	0.819	0.685	0.777	31.454	2.166	1.240	4

Table 4: Performance of classifiers on the Wrist Watches subdatasets

Dataset 1: Wrist Watches datasets								
D1	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.923	0.866	0.866	0.891	20.501	1.232	0.683	1
FCM based	0.915	0.860	0.861	0.882	13.932	1.272	0.706	3
ANN based	0.910	0.873	0.863	0.890	13.728	1.265	0.701	2
SVM based	0.649	0.840	0.622	0.721	37.757	2.429	1.403	4
D2	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.927	0.864	0.863	0.893	13.688	1.250	0.694	1
FCM based	0.923	0.859	0.859	0.887	14.102	1.283	0.712	3
ANN based	0.913	0.872	0.860	0.891	14.042	1.273	0.707	2
SVM based	0.656	0.845	0.634	0.731	36.608	2.401	1.384	4
D3	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.921	0.859	0.862	0.887	13.838	1.235	0.687	1
FCM based	0.911	0.853	0.856	0.877	14.416	1.275	0.710	3
ANN based	0.910	0.873	0.863	0.890	13.728	1.265	0.701	2
SVM based	0.619	0.751	0.595	0.663	40.501	2.533	1.469	4
Average	**Recall**	**Precision**	**Accuracy**	F_1 **measure**	**%Misclassified**	**RMSE**	E	Rank E
FCM+ANFIS based	0.924	0.863	0.864	0.890	13.632	1.239	0.688	1
FCM based	0.916	0.857	0.859	0.882	14.150	1.277	0.709	3
ANN based	0.911	0.873	0.862	0.890	13.833	1.268	0.703	2
SVM based	0.642	0.812	0.617	0.705	38.289	2.455	1.419	4

Table 5: Performance of classifiers on the Jewellery review subdatasets

Dataset 2: Jewellery datasets								
D1	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.923	0.878	0.879	0.897	12.084	1.197	0.659	1
FCM based	0.899	0.875	0.872	0.879	12.844	1.239	0.684	3
ANN based	0.931	0.875	0.880	0.901	11.990	1.208	0.664	2
SVM based	0.888	0.809	0.730	0.824	26.973	1.966	1.118	4
D2	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.921	0.882	0.884	0.898	11.612	1.169	0.643	2
FCM based	0.894	0.880	0.875	0.878	12.508	1.211	0.668	3
ANN based	0.932	0.883	0.887	0.905	11.276	1.147	0.630	1
SVM based	0.933	0.807	0.763	0.845	23.659	1.804	1.020	4
D3	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.930	0.878	0.880	0.901	11.968	1.198	0.659	2
FCM based	0.905	0.872	0.871	0.882	12.880	1.246	0.687	3
ANN based	0.934	0.881	0.883	0.905	11.654	1.168	0.642	1
SVM based	0.949	0.812	0.801	0.864	19.884	1.665	0.932	4
D4	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.987	0.912	0.902	0.948	9.760	1.140	0.619	1
FCM based	0.884	0.878	0.869	0.869	13.086	1.235	0.683	3
ANN based	0.932	0.886	0.889	0.907	11.078	1.139	0.625	2
SVM based	0.948	0.804	0.760	0.848	23.959	1.828	1.034	4
Average	**Recall**	**Precision**	**Accuracy**	F_1 **measure**	**%Misclassified**	**RMSE**	E	Rank E
FCM+ANFIS based	0.940	0.887	0.886	0.911	11.356	1.176	0.645	2
FCM based	0.895	0.876	0.872	0.877	12.830	1.233	0.680	3
ANN based	0.932	0.881	0.885	0.905	11.500	1.165	0.640	1
SVM based	0.929	0.808	0.764	0.845	23.619	1.816	1.026	4

6. Discussion, Conclusion and Future work

The evaluation of customer reviews has become a task of crucial importance to online merchants because they can use this information for planning their future business activities. Existing systems are storing customer reviews without any form of validation, and this increases the uncertainty and noise found in the data.

This paper proposes a novel computational intelligence approach for predicting the numerical review ratings of textual customer reviews using a multi-class rating scale. The proposed framework addresses two important issues: the dimension and imprecision of customer review data. The proposed system uses the Vector Space Model information retrieval technique for indexing each dataset, and applies the Singular Value Decomposition and dimensionality reduction approaches for performing the feature extraction task, which essentially removes noise and reduces the dimensionality of the data. Once the noise is removed, the underlying semantic meaning of each textual review in the dataset is revealed. Thereafter, the Fuzzy C-means and the Adaptive Neuro-Fuzzy Inference System methods are applied to train the system to predict the numerical review ratings of textual reviews using a multi-point rating scale.

An important finding was that, large datasets are not necessarily needed for training classifiers to efficiently predict review ratings. High degrees of

Table 6: Performance of classifiers on the Software review subdatasets

Dataset 3: Software datasets								
D1	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.958	0.860	0.851	0.906	14.882	1.341	0.7450	1
FCM based	0.973	0.843	0.843	0.903	15.720	1.371	0.7641	3
ANN based	0.936	0.874	0.850	0.904	15.022	1.350	0.7499	2
SVM based	0.756	0.845	0.697	0.796	30.287	2.153	1.2279	4
D2	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.955	0.855	0.844	0.902	15.576	1.364	0.7601	1
FCM based	0.973	0.836	0.835	0.899	16.532	1.399	0.7820	2
ANN based	0.929	0.868	0.839	0.897	16.066	1.406	0.7834	3
SVM based	0.732	0.831	0.654	0.770	34.568	2.320	1.3326	4
D3	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.953	0.861	0.851	0.904	14.936	1.333	0.7411	1
FCM based	0.970	0.842	0.841	0.901	15.898	1.370	0.7647	3
ANN based	0.932	0.875	0.850	0.902	15.008	1.358	0.7539	2
SVM based	0.706	0.842	0.659	0.764	34.112	2.298	1.3198	4
D4	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.956	0.862	0.853	0.906	0.147	1.326	0.6640	1
FCM based	0.969	0.845	0.844	0.903	15.582	1.360	0.7579	3
ANN based	0.936	0.875	0.851	0.904	14.872	1.341	0.7447	2
SVM based	0.746	0.843	0.691	0.789	30.928	2.162	1.2359	4
D5	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.962	0.856	0.850	0.906	15.028	1.341	0.7459	2
FCM based	0.973	0.842	0.841	0.902	15.866	1.364	0.7613	3
ANN based	0.939	0.874	0.852	0.905	14.824	1.330	0.7393	1
SVM based	0.946	0.806	0.772	0.865	22.762	1.866	1.0468	4
D6	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.953	0.862	0.854	0.905	14.634	1.321	0.7338	1
FCM based	0.968	0.843	0.843	0.901	15.722	1.358	0.7576	3
ANN based	0.934	0.874	0.852	0.903	14.762	1.321	0.7344	2
SVM based	0.710	0.848	0.670	0.769	32.956	2.231	1.2801	4
D7	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.956	0.856	0.846	0.903	15.412	1.350	0.7522	1
FCM based	0.973	0.837	0.836	0.899	16.448	1.387	0.7758	3
ANN based	0.929	0.870	0.843	0.898	15.744	1.380	0.7687	2
SVM based	0.702	0.841	0.644	0.758	35.579	2.358	1.3567	4
D8	Recall	Precision	Accuracy	F_1 measure	%Misclassified	MSE	E	Rank E
FCM+ANFIS based	0.952	0.856	0.844	0.901	15.578	1.368	0.7617	1
FCM based	0.971	0.838	0.836	0.899	16.440	1.386	0.7752	2
ANN based	0.920	0.872	0.838	0.895	16.231	1.420	0.7910	3
SVM based	0.653	0.841	0.616	0.728	38.442	2.424	1.4042	4
Average	**Recall**	**Precision**	**Accuracy**	**F_1 measure**	**%Misclassified**	**RMSE**	E	Rank E
FCM+ANFIS based	0.956	0.858	0.849	0.904	13.274	1.343	0.7380	1
FCM based	0.971	0.841	0.840	0.901	16.026	1.374	0.7673	3
ANN based	0.932	0.873	0.847	0.901	15.316	1.363	0.7582	2
SVM based	0.744	0.837	0.675	0.780	32.454	2.226	1.2755	4

accuracy were achieved when using subdatasets comprising $10,000$ reviews each. Prior to breaking the datasets into subdatasets, it was important to apply Singular Value Decomposition and dimensionality reduction using the entire dataset, in order to capture the semantic information using a bigger pool of information. This meant that new customer reviews were projected into the dimensional space with high accuracy, which resulted in better prediction performance. Each classifier used for the experiments was separately trained on each subdataset. With this approach, the computational complexity of training the classifiers using a large number of reviews was significantly reduced without compromising their predictive performance.

Experiments were conducted with three large datasets to determine the accuracy of computational intelligence algorithms for predicting customer review ratings. The three datasets were partitioned in subdatasets consisting of $10,000$ reviews, and this resulted in 15 subdatasets. To ensure that the evaluation results were reliable, a 4-fold cross validation was applied on each subdataset. The performance of the proposed predictor yielded high accuracy revealing that when large datasets are concerned, only a fraction of the data (in these experiments $10,000$ reviews were sufficient) is needed for training a system to predict review ratings of textual reviews. The experiment results demonstrate that the proposed FCM+ANFIS approach outperforms some of the common methodologies used in this area. In addition, the neuro-fuzzy predictor module of the proposed framework can be replaced by other computational intelligence predictors such that they can be applied for prediction and classification of reviews on a multi-class scale. In future work, the aim is to improve the performance of the proposed system by replacing some of its components, as for instance the Input Selection Module, with evolutionary optimisation algorithms that could be able to reduce the input dimensionality better than the current approach.

7. References

[1] M. Maity, M. Dass, Consumer decision-making across modern and traditional channels: E-commerce, m-commerce, in-store, Decision Support Systems 61 (2014) 34 – 46.

[2] V. Choudhury, E. Karahanna, The relative advantage of electronic channels: A multidimensional view, MIS Q. 32 (1) (2008) 179–200.

[3] S. Yang, Y. Lu, P. Y. K. Chau, Why do consumers adopt online channel? an empirical investigation of two channel extension mechanisms., Decision Support Systems 54 (2) (2013) 858–869.

[4] K. K. F. Yuen, Toward a ranking strategy for e-commerce products in an e-alliance portal using primitive cognitive network process, Procedia Computer Science 17 (0) (2013) 1091 – 1096, first International Conference on Information Technology and Quantitative Management.

[5] S. Devaraj, M. Fan, R. Kohli, Examination of online channel preference: Using the structure-conduct-outcome framework., Decision Support Systems 42 (2) (2006) 1089–1103.

[6] A. Gupta, B. chiuan Su, Z. D. Walter, Risk profile and consumer shopping behavior in electronic and traditional channels., Decision Support Systems 38 (3) (2004) 347–367.

[7] H. Tang, S. Tan, X. Cheng, A survey on sentiment detection of reviews, Expert Systems with Applications 36 (7) (2009) 10760–10773.

[8] B. Pang, L. Lee, Opinion mining and sentiment analysis, Foundation Trends in Information Retrieval 2 (1-2) (2008) 1–135.

[9] R. Prabowo, M. Thelwall, Sentiment analysis: A combined approach., Journal of Informetrics 3 (2) (2009) 143–157.

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

[11] J. Serrano-Guerrero, J. A. Olivas, F. P. Romero, E. Herrera-Viedma, Sentiment analysis: A review and comparative analysis of web services, Information Sciences 311 (2015) 18 – 38.

[12] M. Koppel, J. Schler, The importance of neutral examples for learning sentiment, Computational Intelligence 22 (2) (2006) 100–109.

[13] M. Gamon, Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis, in: Proceedings of COLING-04, the 20th International Conference on Computational Linguistics, Geneva, Switzerland, 2004, pp. 841–847.

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

[15] R. Moraes, J. F. Valiati, W. P. G. Neto, Document-level sentiment classification: An empirical comparison between svm and ann, Expert Systems with Applications 40 (2013) 621–633.

[16] T. Wilson, J. Wiebe, R. Hwa, Just how mad are you? finding strong and weak opinion clauses, in: In Proceedings of the 19th national conference on artificial intelligence, 2004, pp. 761–769.

[17] H. Saggion, E. Lloret, M. Palomar, Using text summaries for predicting rating scales, in: Proceedings of the 1st Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA), 2010, pp. 44–51.

[18] F. Benamara, C. Cesarano, A. Picariello, D. Reforgiato, V. S. Subrahmanian, Sentiment analysis: Adjectives and adverbs are better than adjectives alone, in: Proceedings of the International Conference on Weblogs and Social Media (ICWSM), 2007, short paper.

[19] B. Pang, L. Lee, Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales, in: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, ACL '05, Association for Computational Linguistics, Stroudsburg, PA, USA, 2005, pp. 115–124.

[20] C. W. K. Leung, S. C. F. Chan, F. Chung, Integrating collaborative filtering and sentiment analysis: A rating inference approach, in: Proceedings of the ECAI 2006 Workshop on Recommender Systems, 2006, pp. 62–66.

[21] M. Ochi, Y. Matsuo, M. Okabe, R. Onai, Rating prediction by correcting user rating bias, in: Web Intelligence and Intelligent Agent Technology (WI-IAT), 2012 IEEE/WIC/ACM International Conferences on, Vol. 1, 2012, pp. 452–456.

[22] G. Ganu, Y. Kakodkar, A. Marian, Improving the quality of predictions using textual information in online user reviews, Information Systems 38 (1) (2013) 1 – 15.

[23] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, ACM Press/Addison-Wesley, 1999.

[24] M. W. Berry, S. T. Dumais, G. W. O'Brien, Using linear algebra for intelligent information retrieval, Tech. Rep. UT-CS-94-270 (1994).

[25] R. B. Cattell, The scree test for the number of factors, Multivariate Behavioral Research 1 (1966) 245–276.

[26] M. Sugeno, T. Yasukawa, A fuzzy-logic-based approach to qualitative modeling 1 (1) (1993) 7–31.

[27] J. S. R. Jang, Anfis: adaptive-network-based fuzzy inference system, Systems, Man and Cybernetics, IEEE Transactions on 23 (3) (2002) 665–685.

[28] C. M. Bishop, Pattern recognition and machine learning, 1st Edition, Springer, 2006.

[29] M. F. Möller, A scaled conjugate gradient algorithm for fast supervised learning, Neural Networks 6 (4) (1993) 525–533.

[30] J. Han, M. Kamber, J. Pei, Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2011.