A Comparison of Fuzzy Approaches to E-Commerce Review Rating Prediction

Giovanni Acampora, Georgina Cosma

School of Science and Technology, Nottingham Trent University, United Kingdom
{giovanni.acampora, georgina.cosma}@ntu.ac.uk

Abstract

This paper presents a comparative analysis of the performance of fuzzy approaches on the task of predicting customer review ratings using a computational intelligence framework based on a genetic algorithm for data dimensionality reduction. The performance of the Fuzzy C-Means (FCM), a neuro-fuzzy approach combining FCM and the Adaptive Neuro Fuzzy Inference System (ANFIS), and the Simplified Fuzzy ARTMAP (SFAM) was compared on six datasets containing customer reviews. The results revealed that all computational intelligence predictors were suitable for the rating prediction problem, and that the genetic algorithm is effective in reducing the number of dimensions without affecting the prediction performance of each computational intelligence predictor.

Keywords: Review rating prediction, rating inference problem, multi-pint scale prediction, Fuzzy logic, Simplified Fuzzy ARTMAP, Fuzzy C-Means, ANFIS.

1. Introduction

The Internet is considered as a channel of communication and an invaluable resource to consumers seeking information about a product or service - it offers consumers the ability to distribute their experiences of products and services in the form of online reviews. The benefits of the Internet to consumers aiming to buy a product is invaluable, because they can access reviews about products from other buyers giving them a perspective on the quality of the product. Product information, provided by people who have already bought and used a product, plays an important role in consumer decision-making on purchases [1], [2]. Consumers generally tend to believe other consumers reviews more than the information posted by sellers, since sellers' tend to disguise negative aspects of the products they are promoting, whereas customers honestly evaluate strengths and weaknesses of the product [3]. Companies use the information obtained from customer reviews to monitor consumer attitudes toward their products/services in real-time mode, and adapt their manufacturing, distribution, and marketing strategies accordingly [4]. Product reviews can be used for revenue forecasting, and early volume of online reviews can be used for early revenue forecasting. Furthermore, companies can use online online review data to generate estimates of their competitors' sales when sale data are not publicly available [4]. This growing number of online product review sites and consequently, the vast amount of information available to companies about their products has attracted research into approaches for data mining.

The activity of rating a product or service using a rating scale is known to as rating-inference within the research area of Opinion Mining and Sentiment Analysis. The most challenging problem is concerned with classifying and predicting review ratings using a multi-point scale. Machine learning (ML) approaches, and in particular 'supervised learning' approaches, have been applied to predict customer review ratings[5],[6],[7], and [8] and have shown to achieve a level of accuracy comparable to that achieved by human experts [9]. Computational intelligence approaches to review rating prediction have been adopted mainly for the binary classification of reviews (i.e. positive or negative) and the challenge still remains on the problem of classifying reviews on a multi-point scale [5], [6], [7]. Although computational intelligence approaches are known to be effective on prediction and classification problems, the computational complexity of applying these algorithms on large data is an issue. The large customer review datasets, and the imprecision and noise found in customer review data adds an extra dimension to the problem, as data must be effectively pre-processed prior to applying computational intelligence algorithms for making the prediction. In the context of customer reviews and review rating prediction, let's define noise as the inconsistency among the textual reviews and the ratings given to reviews. Fuzzy algorithms are well known to deal with imprecise and uncertain data and pose a good solution to the problem of predicting review ratings using a multi-point scale.

The work presented in this paper builds upon our previous work with exposes Fuzzy Logic approaches to addressing the review ratings prediction problem [10],[11]. In particular, we proposed two frameworks suitable for mining large datasets of textual reviews and predicting their numerical ratings on a multi-point scale - the first approach [10] did not include the Genetic Algorithm, whereas the second approach [11] included the Genetic Algorithm, as part of the feature extraction process.

This paper presents a comparative analysis of the performance of the Fuzzy C-Means (FCM), FCM+ANFIS, and the Simplified Fuzzy Adaptive Resonance Theory (SFAM) [12] algorithms using both frameworks [10], [11] in order to determine the effectiveness of Fuzzy algorithms on review rating prediction and the impact of extended feature selection using Genetic Algorithms, as opposed to only using Singular Value Decomposition and dimensionality reduction.

2. The Hybrid Review Rating Predictor Framework

This section summarises the framework (see Fig. 1) proposed by Acampora and Cosma [11], and explains how the framework has been adapted to perform the evaluations with other Fuzzy approaches.

- **The Natural Language Processing Module:** Initially, the predictor takes a dataset of reviews and applies Natural Language Processing (NLP) which includes pre-processing the textual reviews and applying tokenisation and stemming [13]. The NLP Module creates a *Vector Space Model* representation of the pre-processed reviews, creating a term-by-review matrix $A_{m\times n} = [a_{ij}]$, in which each row i holds the frequency of a term, and each column j represents a textual review. Thereafter, the term frequencies of matrix A are further normalised by applying a weighting scheme function [11].

- **The Input Selection Module**: The aim of the Input Selection module is to reduce noise in the data via the use of a statistical technique called Singular Value Decomposition (SVD) and the Genetic Algorithm computational intelligence approach. Initially the SVD is applied to reduce the dimensionality of the data and to remove noise so as to select the most important features useful in characterising the reviews dataset. Thereafter the Genetic algorithm is applied for further feature selection. SVD reduces the dimensionality of data and it is an effective technique for removing noise because it identifies and orders the dimensions of data based on variation which makes it a useful for tasks requiring representation of data in a reduced space. In a customer reviews dataset, noise is present as the inconsistency among the textual reviews and the ratings given to reviews. More precisely, this inconsistency could be between a textual review and a rating given to that review (for example, a positive review receiving a negative rating, or a neutral review receiving a negative rating), and also the inconsistency among textual reviews and their ratings in terms of degree of positivity and negativity which were given the same ratings. 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. Dimensionality reduction is applied to obtain a rank-k approximation to matrix A, where k represents the number of dimensions (or factors) retained, and $k \leq r$. Once SVD and dimensionality reduction are applied, the reviews-by-dimension matrix V is passed into a Genetic Algorithm for further Feature Extraction. Briefly, the Genetic Algorithm is applied to search the further reduced space constructed after SVD, by retrieving the best features for an improved prediction of review ratings. In order to derive the sub-optimal number of features for achieving highest results, the GA takes as input a V_k review-by-dimension matrix where each column represents an input feature vector, a y vector where each element contains the target output of each feature vector, and outputs a set of indices of the selected features that composes the optimum feature set. The vectors with the selected indices (for example, index 2 is the second vector of matrix V) are extracted to form a new reduced reviews-by-dimensions matrix matrix V_{GA}, where $V_{GA} = [v_1, v_2, v_3, \ldots, v_n]$.

- **Neuro-Fuzzy Module**: This module takes as input the matrix V_{GA} and applies a fuzzy learning algorithm to opportunely train the neuro-fuzzy component of the proposed framework. This component can essentially be replaced by any predictor. The focus of the paper is to compare fuzzy predictors and hence the neuro-fuzzy module was replaced by Fuzzy C-Means (FCM) [14], FCM+AFNIS (as described in [11]) and the SFAM algorithms. For a more detailed description of the framework see [11].

2.1. The Fuzzy Predictors

This section provides a summarised description of the Fuzzy C-Means (FCM) [14], AFNIS [15] and the SFAM predictor [12].

2.1.1. Fuzzy C-Means

The Fuzzy C-Means clustering algorithm was proposed by Bezdek [14]. The FCM algorithm takes an input a c number of clusters and a matrix of data to be clustered. FCM returns the 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

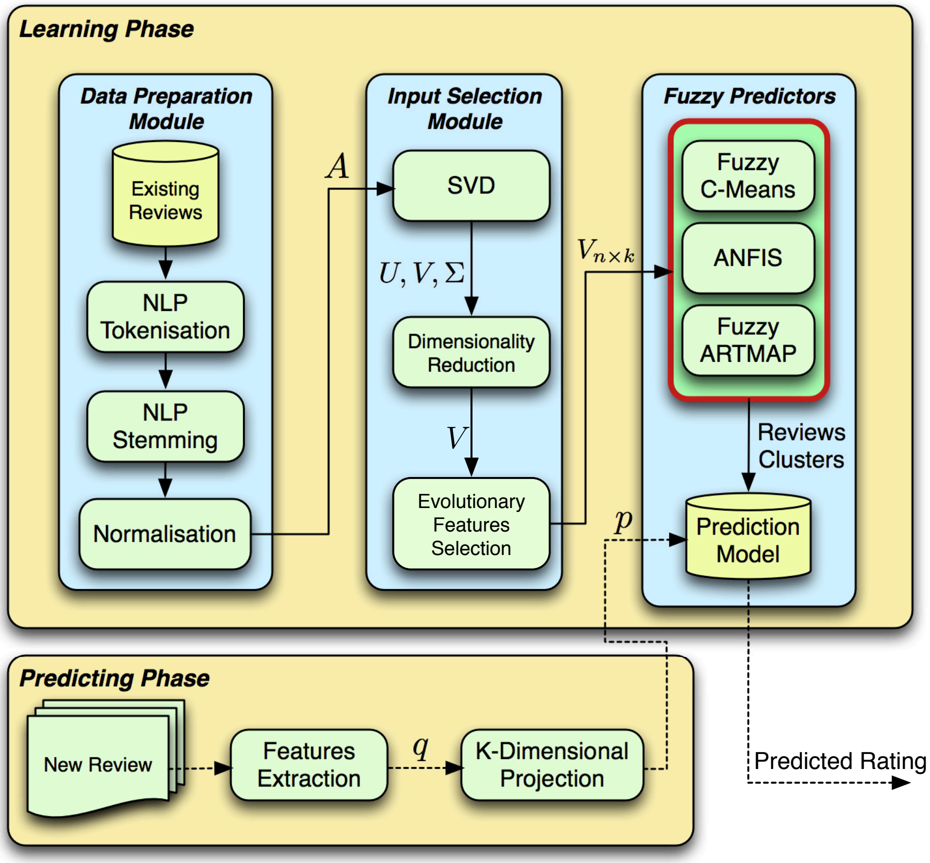

Figure 1: The original review ratings prediction architecture extended with different fuzzy predictors

each element μ_{ik} holds the total membership of a data point (i.e. textual review) v_k belonging to cluster (i.e. a rating class) c_i. FCM updates the cluster centers and the membership grades for each data point (i.e. review), by minimizing an objective function (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).

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

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

FCM clustering generates a Sugeno-type Fuzzy Inference System (FIS) where the number of clusters is equal to the number of rules and membership functions in the generated FIS.

2.1.2. Adaptive Neuro-Fuzzy Inference System

The Adaptive Neuro-Fuzzy Inference System was developed by Jang [15]. ANFIS combines Artificial Neural Networks and Fuzzy Logic. Given a set of input data, ANFIS creates a fuzzy inference system with membership functions generated by adaptive backpropagation learning. In overall ANFIS, consists of five layers [16]. The following description of ANFIS is based on that of [16]. Briefly, in the **layer 1** each node generates a membership grade of a linguistic variable (in the reviews scenario linguistic variables are the rating classes, i.e. class 1, class 2, class n) using a membership function, such as the generalised bell membership function or the Gaussian membership function. In **layer 2** the firing strength of each rule is calculated, and **layer 3** calculates the ratio of each rule's firing strength to the total of all firing strengths. **Layer 4** computes the contribution of each rule toward the overall output, and finally **layer 5** computes the overall output as the summation of the contribution from each rule. During the learning process, ANFIS adapts the parameters associated with the membership functions and tunes them 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. When ANFIS is used in conjunction with FCM, the FIS returned from FCM clustering is fed into the ANFIS and the FIS parameters are tuned using the input/output training data in order to optimise the prediction model. The training process stops once the designated epoch number are reached or the training error goal is achieved and performance is evaluated using the root mean square training errors among the predicted and target outputs.

2.1.3. The Simplified Fuzzy ARTMAP

The Simplified Fuzzy ARTMAP (SFAM) was developed by Kasuba [12] and it is a simplification of the fuzzy ARTMAP originally proposed by Carpenter et al. [17]. The SFAM is also known as Predictive ART and can only be applied to solve classification tasks, has reduced computational overhead and architectural redundancy (i.e. simplified learning equations, a single user selectable parameter and uses a small number of iterations to learn all training patterns) [18]. The SFAM is a neural network classifier which comprises of an input and an output layer. The architecture of the SFAM can be described as follows:

Let V_{GA} be the reduced review-by-dimension matrix passed into the *Raw Input Layer* which contains a *complement coder* which normalises the input matrix and it represents the presence or absence of a particular feature in the input pattern. The weights (w) from each of the output category nodes reach down to the input layer. The category layer simply contains the names of the M number of categories that the network has to learn (i.e. number of review rating classes). Once the complement coded representation of matrix V_{GA} is constructed, all output nodes become active to some degree. The next step is to determine the winning node. Let I be the complement coded representation of matrix V_{GA} then the output activation is denoted by T_j for the jth output node, where W_j is the corresponding weight vector given by function (4):

$$T_j(I) = \frac{|I \wedge W_j|}{\alpha + |W_j|}. \tag{4}$$

The value of α is kept as a small value close to 0. The node with the highest output activation value is the winning output node,

$$Winner = max(T_j(I)) \tag{5}$$

The winner output is basically the category assigned to an input vector (i.e. the predicted rating of a review represented as an input vector). Sometimes, more than one T_j is maximal, and when this occurs the output node which has the smallest index value is selected as the winner output.

The vigilance parameter and match tracking mechanism are used for training the network. A vigilance parameter p, 0<p<1, controls the granularity of the output node encoding. Initially, p is equal to its baseline value $\overline{p}$. High vigilance values make the output node much fussier during pattern encoding, and low vigilance renders the output node to be liberal during the encoding of patterns. During the learning phase, the *match tracking* mechanism adjusts the vigilance parameter, p, based on the classification errors. The match function shown in 6 determines whether learning should occur

$$\frac{I \wedge W_j}{|I|}. \tag{6}$$

Basically, the vigilance parameter and the match function value determine whether the current input is a good enough match to a particular output node, or whether a new output node should be formed to encode the input pattern.

3. Evaluating Product Reviews

3.1. Methodology

This section describes the results from experiments carried out for testing the performance of Fuzz approaches on the task of predicting review ratings using the hybrid computational intelligence framework proposed by [11]. In addition, we evaluate the impact of applying a Genetic Algorithm on the review rating performance of classifiers using the framework proposed by [10] which does not include additional feature extraction via Genetic Algorithms. Each framework, with GA [10] and without GA [11], was evaluated using six datasets, and three fuzzy-based approaches. Initially the NLP module (see Section 2) was applied in order to preprocess each dataset in order to obtain a normalised term-by-review matrix A. This matrix A was then input to the *Input Selection* module which resulted in SVD being applied to the term-by-review matrix A with the Dimensionality Reduction parameter set to k dimensions, where k was selected experimentally [11]. Dimensionality Reduction was performed by using k dimensions and the derived review-by-dimension matrix $V_{n \times k}$ where n is the total number of reviews and k is the number of features, was used as input into each of the Fuzzy algorithms. For comparisons without GA, the $V_{n \times k}$ matrix was passed into a Fuzzy predictor as described in [10], whereas for comparisons with GA, the $V_{n \times k}$ matrix was passed into the Genetic Algorithm which reduced the matrix further and returned a V_{GA} matrix which was then passed into the Fuzzy predictor as described in [11]. The output of all predictors is a set of values, which comprised one numerical rating for each review.

3.2. Evaluation Measures

This section presents the evaluation measures adopted for assessing the performance of each classifier on predicting the rating classes of reviews. Performance was measured using the *Accuracy* and the *Percentage of Misclassification* evaluation measures. *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* is also a reliable measure because it gives the percentage of reviews which have been misclassified. Misclassification is computed as 1-Accuracy, and the lowest the value the better the performance of the classification system. The functions of accuracy, acc, and misclassification, mc, are shown in functions (7) and (8) respectively.

$$acc = \frac{|TP| + |TN|}{|P| + |N|} \tag{7}$$

$$ms = \frac{|FP| + |FN|}{|P| + |N|} \tag{8}$$

where,

- 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 $|N| = |FP| + |TN|$.

Sections 3.3 and 3.4 describe the results of the experiments. The performance values provided are those returned from checking (i.e. testing) the performance of each predictor by means of predicting the numerical ratings of reviews after the training process is performed.

3.3. Analysis of the Cornell Datasets

The Cornell corpus[1] contains four separate datasets of movie reviews, where each dataset was written by the same author. The Cornell corpus has been adopted by several researchers conducting sentiment-analysis experiments and for this main reason it was decided to include these datasets in the comparison [6], [19]. These reviews are classified on a 4-point numerical rating scale. Table 1 shows the number of terms and reviews in each dataset; and Table 2 shows the number of reviews in each rating category. The results of the experiments conducted on all four datasets are shown in Table 3.

Cornell Movie Review Datasets		
Dataset	**No. of terms**	**No. of reviews**
Dataset 1	1,616	1,770
Dataset 2	1,638	1,307
Dataset 3	8,917	902
Dataset 4	7,104	1,027
Total	19,275	5,006

Table 1: Dataset Characteristics

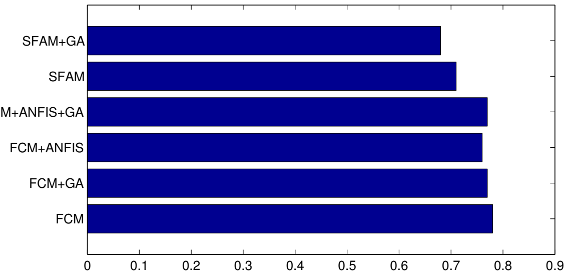

Figure 2: Cornell Datasets Performance

The results of the Cornell dataset experiments show that halving the number of dimensions does not really affect the predictors performance. Table 7 shows the differences between the accuracy of the predictors with and without GA. The results of the comparison revealed that, on average across all Cornell datasets, a slight and insignificant decrease in the FCM and SFAM predictors performance of 0.01 and 0.03 decrease when GA was applied. The performance of FCM+ANFIS improved by 0.01 when GA was applied. However, these values are very small and this may not have been the case if another dataset was used. Importantly, the number of dimensions were halved after GA was applied and the performance of the predictors was not comprised, since performance was very close before and after GA was applied. The results of the comparison shows that applying GA for further feature selection (i.e. after SVD) has a great positive impact on the complexity of applying a Fuzzy or any other computational intelligence predictor.

It is important to point out that although these datasets have been widely used in sentiment analysis and opinion miming experiments, one issue related to these datasets is that each dataset was written by the same author, which means that it is likely that less noise will exist in the dataset. Nevertheless, the results clearly show that the performance

[1] http://www.cs.cornell.edu/people/pabo/movie-review-data/

	Dataset 1		Dataset 2		Dataset 3		Dataset 4	
Rating Class	Count	Percent	Count	Percent	Count	Percent	Count	Percent
1	192	11.79	139	11.56	115	12.75	172	17.65
2	527	30.72	293	23.34	295	32.71	441	43.84
3	767	44.28	597	46.60	334	37.03	303	30.41
4	288	17.21	282	22.50	158	17.52	115	12.10

Table 2: Cornell Dataset Ratings

Cornell Datasets							
		Without GA (k=20)			**With GA (k=10)**		
		FCM	FCM+ANFIS	SFAM	FCM+GA	FCM+ANFIS+GA	SFAM+GA
Dataset 1	acc	0.78	0.78	0.71	0.78	0.78	0.66
	ms	0.22	0.22	0.29	0.22	0.22	0.34
Dataset 2	acc	0.80	0.78	0.73	0.79	0.80	0.74
	ms	0.20	0.22	0.27	0.21	0.20	0.26
Dataset 3	acc	0.73	0.72	0.65	0.71	0.70	0.64
	ms	0.27	0.28	0.35	0.29	0.30	0.36
Dataset 4	acc	0.79	0.77	0.75	0.79	0.80	0.68
	ms	0.21	0.23	0.25	0.21	0.20	0.32
Average	acc	0.78	0.76	0.71	0.77	0.77	0.68
	ms	0.22	0.24	0.29	0.23	0.23	0.32

Table 3: Cornell Movie Review Dataset Results

of each predictor before and after GA is very close meaning that by applying GA the dimensionality of data is reduced making the predictor more efficient. Finally, the results show that the prediction framework can be applied to predict review ratings on small datasets containing reviews written by the same author and GA can be applied to efficiently reduce the dimensionality of the data.

3.4. Analysis of the Stanford Datasets

A second experiment was conducted using two large datasets[2], namely a software reviews dataset and a wrist watch reviews dataset. The software reviews dataset consists of 95,084 textual reviews on software products, and the wrist watches reviews dataset consists of 68,355 textual reviews of wrist watches. Each review is accompanied by a numerical rating provided by the person who provided the review. The reviews in the software and wrist datasets were written by various authors, as opposed to the Cornell datasets where each dataset was written by a single author. Table 4 shows the number of terms and reviews in each dataset; and Table 5 shows the number of reviews in each rating category. The results of the experiments conducted on all datasets are shown in Table 6.

Stanford Review Datasets		
Dataset	**No. of terms**	**No. of reviews**
Wrist Watches	16,547	68,355
Software	11,210	95,084
Total	27,757	163,439

Table 4: Stanford Dataset Characteristics

	Wrist watches		Software	
Rating Class	Count	Percent	Count	Percent
1	5376	7.86	25798	27.13
2	3650	5.34	8076	8.49
3	5398	7.90	8894	9.35
4	14725	21.54	16360	17.21
5	39206	57.36	35956	37.81

Table 5: Stanford Dataset Ratings

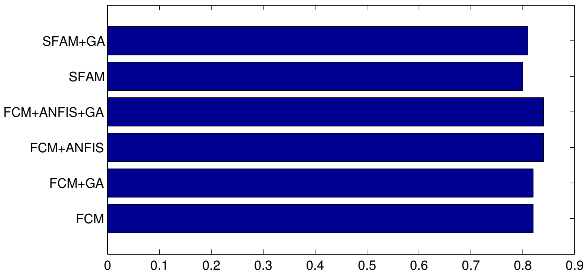

Figure 3: Stanford Datasets Performance

As shown in Table 6, using the wrist watches dataset, the performance of the FCM and FCM+ANFIS predictors was the same with and without GA (0.89 for each predictor). The performance of SFAM was improved by 0.03 after GA was applied. On the software dataset, GA did not have any impact on the accuracy of the FCM and SFAM predictors(0.75 and 0.77 respectively),

[2] Available at http://snap.stanford.edu/data/web-Amazon.html

Stanford Datasets							
		Without GA (k=40)			With GA (k=35)		
		FCM	FCM+ANFIS	SFAM	FCM+GA	FCM+ANFIS+GA	SFAM+GA
Wrist Watches	acc	0.89	0.89	0.83	0.89	0.89	0.86
	ms	0.11	0.11	0.17	0.11	0.11	0.14
Software	acc	0.75	0.79	0.77	0.75	0.78	0.77
	ms	0.25	0.21	0.23	0.25	0.22	0.23
Average	acc	0.82	0.84	0.80	0.82	0.84	0.81
	ms	0.18	0.16	0.20	0.18	0.16	0.19

Table 6: Stanford Dataset Results

whereas the performance of the FCM+ANFIS very slightly decreased, by less than 0.01 (rounded up as 0.01). In overall across the Stanford datasets, best performance was achieved by FCM+ANFIS(0.84) and FCM+ANFIS+GA(0.84), closely followed by FCM(0.82) and FCM+GA(0.82), SFAM+GA(0.81) and SFAM (0.80).

In overall, as shown in Table 7 the GA slightly improved the performance of FCM+ANFIS (by 0.01) and slightly decreased the performance of the SFAM (by 0.01). This finding clearly shows the benefits of including GA for further feature extraction. On large datasets, reducing the number of dimensions reduces computational complexity and the time needed to train a predictor, since it will be trained in a smaller matrix. The fact the performance of each predictor was not affected by adding GA is a positive result clearly demonstrating the improved efficiency of the prediction model when GA is included to reduce the dimensionality of data.

4. Conclusion

The growing number of online product review sites and consequently, the vast amount of information available to companies about their products has attracted research into approaches for data mining. This paper provides a comparative analysis of the performance of fuzzy approaches on the task of predicting review ratings in e-commerce using a hybrid computational intelligence framework. The performance of the Fuzzy C-Means (FCM), a neuro-fuzzy approach combining FCM and the Adaptive Neuro Fuzzy Inference System (ANFIS), and the Simplified Fuzzy ARTMAP (SFAM) was compared on six datasets containing customer reviews. The results revealed that the Genetic algorithm is effective in reducing the number of dimensions without affecting the prediction performance of each algorithm.

These results clearly show the benefits of performing GA, in addition to the SVD algorithm, for improving feature extraction. On large datasets, reducing the number of dimensions reduces computational complexity and the time needed to train a predictor. Since the performance of each predictor was not affected by adding GA, clearly demonstrates the improved efficiency of the prediction model when GA is included. In overall, the differences in performance among the predictors was very small and hence we cannot confidently say that once approach is better than another, although on large datasets the FCM+ANFIS+GA appears to be the best approach based on the experiments presented in this paper and past experiments [10], [11].

In future work we plan to perform more experiments using other large datasets in order to determine confidently which predictor is better for the review rating prediction problem. In addition, we plan to continue our work on feature extraction by exploring the suitability of memetic algorithms on the review ratings prediction problem.

References

[1] Beverley A. Sparks and Victoria Browning. The impact of online reviews on hotel booking intentions and perception of trust. *Tourism Management*, 32(6):1310 – 1323, 2011.

[2] A. Papathanassis and F. Knolle. Exploring the adoption and processing of online holiday reviews: A grounded theory approach. *Tourism Management*, 32(2):215–224, 2011. cited By (since 1996)18.

[3] Do-Hyung Park, Jumin Lee, and Ingoo Han. The effect of on-line consumer reviews on consumer purchasing intention: The moderating role of involvement. *Int. J. Electron. Commerce*, 11(4):125–148, July 2007.

[4] Chrysanthos Dellarocas, Xiaoquan (Michael) Zhang, and Neveen F. Awad. Exploring the value of online product ratings in revenue forecasting: The case of motion pictures. *Journal of Interactive Marketing*, 21(4):23–45, 2007.

[5] Huifeng Tang, Songbo Tan, and Xueqi Cheng. A survey on sentiment detection of reviews. *Expert Systems with Applications*, 36(7):10760–10773, sep 2009.

[6] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundation Trends in Information Retrieval*, 2(1-2):1–135, January 2008.

[7] Rudy Prabowo and Mike Thelwall. Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2):143–157, 2009.

[8] Moshe Koppel and Jonathan Schler. The importance of neutral examples for learning sen-

Cornell Datasets			
	FCM vs FCM+GA	FCM+ANFIS vs FCM+ANFIS+GA	SFAM vs SFAM+GA
Dataset 1	0.00(=)	0.00(=)	0.04(>)
Dataset 2	0.01(>)	-0.02(<)	-0.01(<)
Dataset 3	0.02(>)	0.01(>)	0.01(>)
Dataset 4	0.00(=)	-0.03(<)	0.07(>)
Average	0.01(>)	-0.01(<)	0.03(>)
Stanford Datasets			
Dataset 1	0.00(=)	0.00(=)	-0.03(<)
Dataset 2	0.00(=)	0.00(=)	0.00(=)
Average	0.00(=)	0.00(=)	-0.01(<)
Mean Average	0.00(=)	-0.01(<)	0.01(>)

Table 7: Table shows difference in accuracy when GA was and was not applied. The difference is computed by a subtraction, $diff = acc\_non\_GA - acc\_with\_GA$. To ease readability, next to each result, a symbol indicates the type of difference: = is no difference, > is non-GA returned higher accuracy, < is GA returned higher accuracy.

timent. *Computational Intelligence*, 22(2):100–109, 2006.

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

[10] G. Acampora, G. Cosma, and T. Osman. An extended neuro-fuzzy approach for efficiently predicting review ratings in e-markets. In *Fuzzy Systems (FUZZ-IEEE), 2014 IEEE International Conference on*, pages 881–887, July 2014.

[11] Giovanni Acampora and Georgina Cosma. A hybrid computational intelligence approach for efficiently evaluating customer sentiments in e-commerce reviews. In *Intelligent Agents (IA), 2014 IEEE Symposium on*, pages 73–80, Dec 2014.

[12] T Kasuba. Simplified fuzzy artmap. *AI Expert*, 8(11):18–25, 1993.

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

[14] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1981.

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

[16] Jyh-Shing Roger Jang. Input selection for anfis learning. In *Proceedings of the IEEE International Conference on Fuzzy Systems*, pages 1493–1499, 1996.

[17] G. Carpenter, S. Grossberg, and D. Rosen. Fuzzy art: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks*, 4(6):759 – 771, 1991.

[18] Boonruang Marungsri and Suphachai Boonpoke. Applications of simplified fuzzy artmap to partial discharge classification and pattern recognition. *WTOS*, 10(3):69–80, March 2011.

[19] Bo Pang and Lillian 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, pages 115–124, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.