

## Mining: Students Comments about Teacher Performance Assessment using Machine Learning Algorithms

Guadalupe Gutiérrez<sup>1</sup>, Juana Canul-Reich<sup>2</sup>,  
Alberto Ochoa Zezzatti<sup>3</sup>, Lourdes Margain<sup>1</sup> and Julio Ponce<sup>4</sup>.

<sup>1</sup>Universidad Politécnica de Aguascalientes,

<sup>2</sup>Universidad Juárez Autónoma de Tabasco,

<sup>3</sup>Universidad Autónoma de Ciudad Juárez,

<sup>4</sup>Universidad Autónoma de Aguascalientes.

**Abstract.** The exponential expansion of sentiment analysis is mainly due to the interest of companies to obtainment users' views about products and services. Likewise, sentiment analysis has been widely and successfully used in publicity and selling strategies as well as behaviour patterns and user preferences identification. In the same way, has been applied to educational areas to increase the learning, and e-learning quality through the mining analysis data on teaching performance assessment or student's comments. This work describes the development and evaluating the process of a Model called "*SocialMining*", which is focused on higher education with the purpose of improve teaching techniques of teachers and recommend courses for teacher improvement, through teacher performance evaluation made by students comments. The SocialMining model analyze students' comments by means representative machine learning algorithms such as: Support Vector Machines and Random Forest. We applied metrics as evaluation measure to measuring the performance of algorithms. Finally, we have implemented this model in Universidad Politécnica de Aguascalientes (Mexico) and the results obtained show that it is feasible to perform sentiment analysis to classify comments of Teacher Performance Assessment using Machine Learning with high accuracy (85%).

**Keywords:** Sentiment analysis, Teacher Performance Assessment, Machine learning, Classification of comments.

### 1. Introduction

Inducing knowledge to students is a difficult task for teachers, who need to consider several factors, such as the cognitive ability of each student, distracting behaviours and especially learning methods, all these factors can influence on the relative level of students in class session. In educational institutes, the performance evaluation of each professor is a relevant issue for quality measures. Nevertheless, this process has a high complexity by resulting from the huge quantity of criteria needed to analyze before submitting a result to the teacher. The literature review [1], [2] suggest tools for teacher's evaluation such as guides, rubrics with different evaluation criteria or evaluation in a class by specialized academics in pedagogy area. However, when the teacher performance evaluation is based on comments from students, is quite crucial to use advanced methods to analyze quickly the amount of data that can be generated. Comments from students can give the best feedback to the teacher and suggest courses of teacher improvement, in accordance with the weaknesses detected by students of the course, with the purpose of increasing the academic quality.

In this work, we used a data collection from students of the Universidad Politécnica de Aguascalientes (UPA), based on a teachers performance survey, applied through a computational system. This survey (named SED), is applied two times per each scholar period of four months. The SED contains 20 multiple-choice questions and one open question asking student opinion about their teachers. Once ended the SED opening time, the results are exports to a database that can contain around twelve thousand comments depending on the quantity of teachers and students working during the SED application period. The Department of Overcoming Teaching considers the results of the first 20 question to generate a report distributed within each Academic Program Director (APD), but there is not a specific analysis of students' comments yet. Therefore, each APD has to read them and take the correspondent decisions.

Nowadays, the teacher's performance evaluation has been used to measure the performance of a professor, indicating if he is a good or bad professor. Notwithstanding, it's important to make the focus on students opinions analysis to detect particular situations where is possible recommend academic courses or workshops to ameliorate the teaching abilities of the professors, as well as offer the necessary support of psychological specialist-oriented to education in the proper situations. Due to the huge quantity of comments that can be generated through the teacher's performance evaluation, could be complicated to detect if any student or group with certain problems require immediate attention. In addition, as the survey is applied just two times per scholar period, there is no way to assure the early detection of potential problems with teachers or students. By this reason, and based on the good results found in the literature [3]–[5], it is proposed to use Twitter to make comments about teachers. Nonetheless, a tool to analyze the comments created during the scholar period is also required.

Hence, based on the problems described, the development of a sentiment analysis model to support the teachers' performance evaluation, is considered feasible, with the principal objective of favouring the improvement of teaching process through the semi-automation analysis of comments students. The Figure 1 presents the SocialMining Model based on the proposed objective, perform a classifier of comments by machine learning algorithms based on a process of features selection and a process of setting parameters (parameter optimization process).

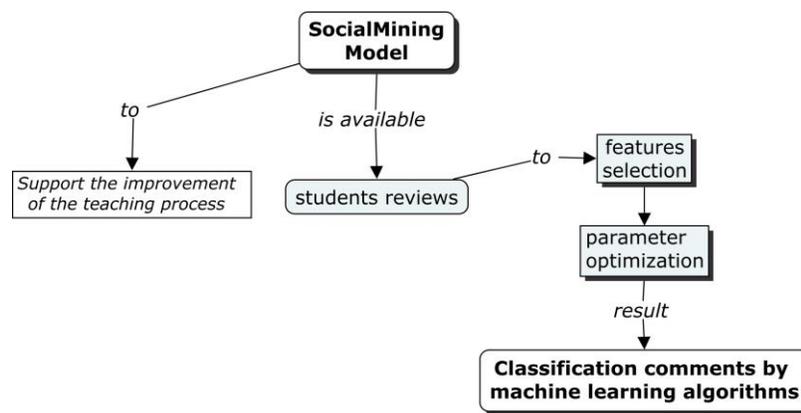


Figure 1. General diagram of SocialMining Model process.

## 2. Related Work

Sentiment analysis also called opinion mining, has been defined as an area of computational study of opinions, feelings and emotions expressed in texts [6]. This area arises on business intelligence, but now is extended to other areas such as politics, medicine, education, among others. Some authors [7], [8] argue that most of the time individuals seek opinions before making a decision, so that the information resulting from a sentiment analysis process allows identify potential customers for a company, the level of satisfaction of a product or service, or the success or dislike of users for a political campaign. Researchers of sentiment analysis have applied different approaches to predict the sentiment in a phrase or word. In Education, sentiment analysis has been combined with natural language process, unsupervised learning and machine learning. Related works described below have been successful in their different combinations of methods and algorithms. For example, literature reports the use of algorithms like SVMs, Naïve Bayes to complement Naive Bayes in the Development of an application to know the student emotional state, in this study the students are reviews at the University of Portsmouth [3]. At the same time, concepts as lexicon based, machine learning and hybrid approaches, were used in the development of an application called SentBuk to retrieve identify users sentiment polarity and emotional changes [9]. Others works focus different aspects a) on evaluation from student opinions about teachers using Lexicon in Thai, SVMs, ID3 and Naive Bayes. Reviews by students are from Loei Raja hat University [10]. b) Design of an experimental study to predict teacher performance.

Lexicon with 167 keywords positive and 108 keywords negative. The dataset has 1,148 feedbacks by students, obtained from RateMyProfessors.com [11]. Some authors proposed a method to detect the feeling of students

on some topics and support the teacher to improve their teaching process. The algorithm used is Latent Dirichlet Allocation, SVMs, Naive Bayes and Maximum Entropy. In the data set are used Movie reviews [12] and comments by 75 students collected from Moodle [13]. Finally, other related works explain about a system to help the developer and educator to identify the most concentrated pages in E-learning portals using Bayesian classification, Naïve Bayesian and SVMs and use a collection of 100 users review from the website Functionspace.org [5]. By other way, implement the sentiment analysis technique in Mlearning System [14], enable know the user opinions of the M-learning system, using Naive Bayes, KNN and Random Forest and 300 reviews from www.market.android.com From this section, we can see that the most of the previous research has focused on particular aspects of the Education. In this work, we proposed a model to evaluate teacher performance consider Spanish reviews from students and applying machine learning algorithms, to classify into positive, negative and neutral polarity.

### 3. SocialMining Model Architecture

The SocialMining Model implements two machine learning algorithms (SVMs and Random Forest) to perform the classification process of teacher evaluation comments. Figure 2 shows the architecture of the proposed model.

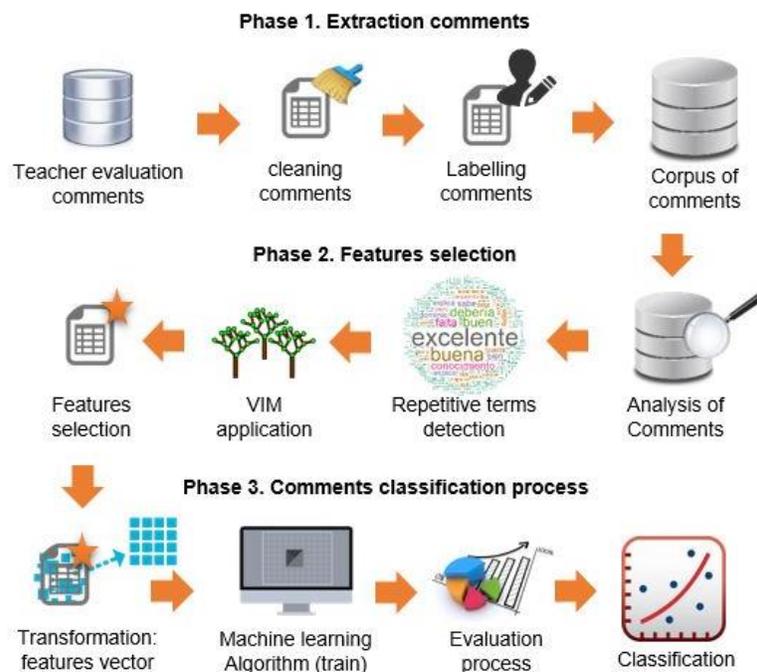


Figure 2. SocialMining Model Architecture

**Phase 1. Generation of the corpus of comments.** In this phase, it is generated a corpus of comments, which are extracted through Teacher Evaluation System of the UPA (SED) and Twitter, where students use hashtags to identify teachers. To create the corpus of comments, it is considered to include only those comments that are free of spam (characterized in this study as texts with strange characters, empty spaces or comments unrelated to teacher evaluation). After obtaining the comments of teacher evaluation, it is carried out a process of manual tagging conducted by teachers of the institution, who consider a range of values to classify comments. Negative comments are labelled within the range of -2 to -0.2, using the value -2 for those very negative comments. Comments labelled within the range of +0.2 to +2 are considered positive, the value +2 is used to label very positive comments. Moreover, neutral comments are labelled with a value equal to 1. Once the process of tagging on comments, we proceed to perform a cleaning process, in which the stop words and nouns irrelevant

are removed (e.g. teacher, college, class, subject, school). Similarly, each of the terms of comments is converted to lowercase and punctuation and numbers are deleted.

**Phase 2. Feature selection.** In this phase is performed the features selection, which is used to train the SVMs and Random Forest algorithms. The Random Forest Importance Measure method (VIM) is used for feature selection. Figure 3 shows the feature selection process.

**Phase 3. Comments Classification process.** In this phase, the corpus of comments is partitioned into two datasets. One of the datasets is dedicated to the process of training, and it is implemented with SVMs and Random Forest to find patterns or relationships among data; the second dataset is considered for the test procedure in order to adjust the performance model (once the algorithm is trained, called model). In this work, we used two-thirds of comments dataset for train and one-third for the test.

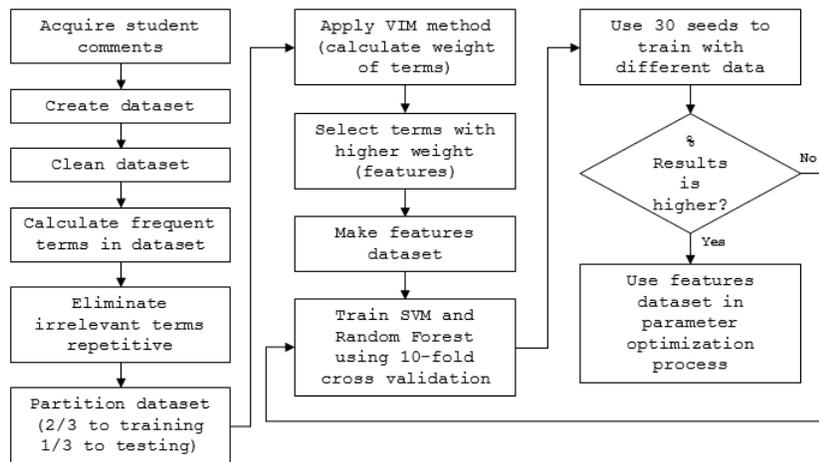


Figure 3. Proposed algorithm for feature selection.

The machine learning algorithms implemented in the SocialMining Model are SVMs (kernels: linear, radial basis and polynomial) and Random Forest. To improve the performance of these algorithms, a parameter optimization process is applied. In parameter optimization process, we used 10-fold cross-validation. Cross-validation divides the dataset into k independent new datasets (k folds).

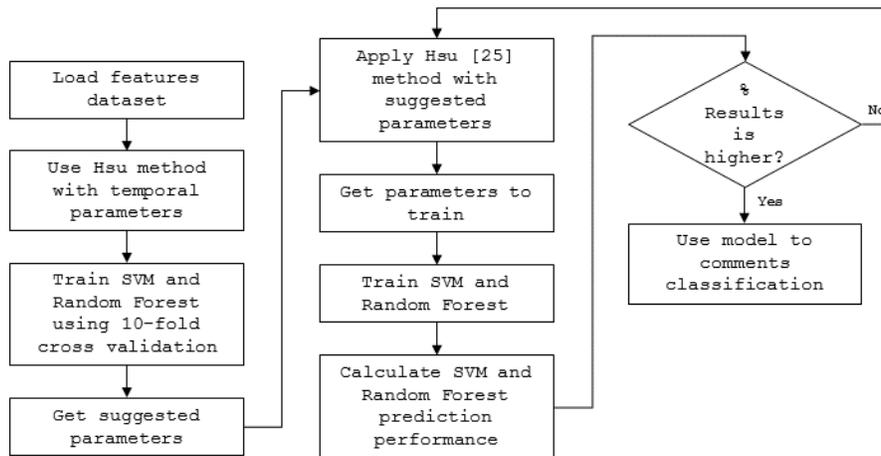


Figure 4. Proposed algorithm for setting parameters.

Then, it performs k loops in which k-1 partitions of the dataset is used for training and the rest for testing. When all the k loops have terminated, the cross-validation accuracy is calculated. Through the parameter optimization process, it is possible to identify identity Whether the value of each parameter remains constant or varies. Figure 4 shows the diagram of parameter optimization process in SVMs and Random Forest. Once completed the setting parameter process, we proceed to perform the classification of teacher evaluation comments. This process uses typical performance measures in machine learning to evaluate the performance.

#### 4. Data and Methods

##### Data collection.

The data used in this study was obtained by SED and Twitter, where three groups of the Engineering Systems career evaluated 21 teachers of the first school term of 2016. By this evaluation, a total of 1,040 comments were collected, which formed the corpus of comments used in this work. Subsequently, a labelling process is performed, where 515 negative comments and 515 positive comments were identified, as well as 10 neutral comments. Based on the number of positive and negative comments, the comments corpus is considered a balanced corpus, because the priority is to perform a binary classification (sort comments in classes: positive and negative). Also, the “average value in comments” column in the table, shows values into the positive and negative class are equitable. See table 1.

**Table 1.** Quantity and average value in comments

	Comments numbers	Average value in comments
Positive comments	515	1.94
Negative comments	515	-1.94

For the experimentation process, two third parts of the corpus were used to shape the training set, while the other third part is confirmed as a testing set. 30 executions were done applying the 10-fold cross validation method to the SVMs and Random Forest algorithms training. In each execution a different seed was controlled, caring for the equality of seeds and the hold on the algorithm. The use of different seed in each execution ensure that the corpus partition will be different. The methods used in this work are described in the next section.

##### Methods.

Support Vector Machines (SVMs) is a supervised learning algorithm, focused as a predictive multiclass classifier suggested for the first time by Vapnik [15]. SVMs have been used as a sentiment polarity classifier in reviews with high-performance results [3], [10], [16]. This algorithm works with the next kernels: linear, Gaussian radial basis, polynomial and sigmoidal. However, in this work we use the first three kernels to classify comments of Teacher Performance Evaluation, table 2 shows the parameters and equations of each kernel.

**Table 2.** Support Vector Machines: Kernels

Kernel	Parameter	Equation
Linear	C	$K(x, y) = X^T y + c$
Gaussian radial basis	C, $\gamma$	$K(x, y) = \exp(-\gamma \ x-y\ ^2),$ $\gamma > 0$
Polynomial	C, degree, coef	$K(x, y) = (x^T y + \text{coef})^d$

- In linear kernel equation, X and Y are vectors in input space, and C is the penalty parameter.
- In Gaussian radial basis, X and Y are vectors in input space, controls the standard deviation.
- In polynomial kernel equation, X and Y are vectors in input space and d is the polynomial grade.

In this work, we make a parameter optimization process for each kernel to improve the SVMs performance in comments classification.

Random Forest is an ensemble learning method focused on predictive algorithm introduced for the first time by Breiman [17]. This training arises from bagging and CART-split [18] and is based on sets of decision trees, where each trained set brings a predictive output which allows the generation of more sets oriented to the most popular class, adding the advantages of stability, precision and robustness to the tuning process knowing as an overfitting due to the continued learning of characteristics of themselves by the overtraining. This classifier consists of a collection of structured tree classifiers:

$$[h(x, \theta_k), k = 1, \dots]$$

Where the random vectors  $[\theta_k]$  are independent identically distributed and each tree generate a unit vote for the most popular class at input  $x$ .

The high performance in classification process of Random Forest is caused by the regulation of the two following parameter:  $[mtry \in [1, \dots, p]]$  random input variables of each division and  $[ntree]$  trees quantity. This algorithm has good performance in textual comments classification [14], [19]. Variable Importance Measures is a features filtering method derived from Random Forest, which rank candidate predictors such as nominal or metric responses that are not proven appropriate for ordinal response. The main purpose is to evaluate the importance of a variable  $[x_m]$  for predicting  $Y$  by adding to all the nodes  $t$  the weighted impurity decreases  $a$ , where  $X_m$  is used over all the trees in the forest  $NT$ .

$$imp(X_m) = \frac{1}{N_T} \sum_T \sum_{t \in T: v(S_T)} P_t \Delta i(S_t, t)$$

Where:  $p(t)$  is the proportion  $NT / N$  of samples accomplishing  $t$ ,  $v(S_T)$  is the variable used to split  $ST$  In this way, VIMs allows optimization of features selection to ameliorate the performance of the machine learning algorithms for classifications. According to others author [20], this method gets better results comparing to KNN and SVM.

### Methods used to measure performance.

We used typical performance measures in machine learning such as: accuracy, balanced accuracy, sensitivity, specificity and ROC space [21]. Confusion Matrix. The Confusion Matrix introduced by Kohavi and Provost [22] is a method used to represent the classification performance of machine learning algorithms for attributes selection, see table 3. The matrix contains the comparison between the classification results of a predictive model and the real classification values. The rows of the following table consider the predictive values, while the columns characterize the real values.

**Table 3.** Confusion matrix

	Real value	
Predictive value	TP	FP
	FN	TN

Where: TP as True Positive, FP as False Positive, FN as False Negative and TN as True Negative.

Accuracy (ACC). Evaluate the predictive model performance in binary classifications, calculating its exactitude dividing the number of correct cases identified over the total cases.

$$ACC = \frac{TP + TN}{TP + FN + FB + TN}$$

Where: TP as True Positive, TN as True Negative, FN as False Negative and FP as False Positive.

Balanced Accuracy (WACC). For data sets of unequal distribution of the two classes, the WACC produce a better estimate classifier performance for both sets

$$WACC = \frac{\frac{TP}{TP + FN} + \frac{TN}{FP + TN}}{2}$$

Where: TP as True Positive, TN as True Negative, FN as False Negative and FP as False Positive.

Sensitivity (TPR), is a statistical metric also called Recall, represents the fraction of TP instances over the total instances (TP+FN) to identify those True Positive who are correctly classified.

$$TPR = \frac{TP}{TP + FN}$$

Where: TP as True Positive and False Negative

Specificity (FPR), it is a statistical metric to identify those True Negative classified in a correct manner.

$$FPR = \frac{TN}{TN + FP}$$

Where: TP as True Negative and False Positive

ROC space. Receiver Operating Characteristics (ROC) graphs are a useful technique to measure the classifiers performance and visualize it in a two-dimensional plane. The area under the ROC curve enables the performance measurement, getting results oscillating between 0 and 1, where 1 represents a perfect classification and is considered as the point nearest to the curve. Its functionality is based on the ratio representation by TPR and FPR. Hence the graphs can show diverse figures depending on the sensitive and specificity values.

Cohens kappa, is a statistic parameter that measures inter-annotator agreement, it adjusts the prediction of the real class focusing 1.0 as the complete setting. It is generally considered as a more robust measure due to agreement occurring by chance than simple percent agreement calculation [23].

## 5. Experimentation and Results

### Feature selection process.

For carrying out the process of selecting features the VIM method is applied, which uses the Out Of Bag (OOB) method to obtain different samples of features and measure the accuracy of prediction [24]. Once the corpus of comments has been submitted through a cleansing process, it is reviewed most of the terms that contribute to the classification process. Through the generation of a word cloud is possible to visualize terms with more frequency in the corpus. Figure 7 shows the word cloud generated, in which it is possible to observe some terms that must be eliminated (e.g. 'ejercicios', 'nivel', 'maneja', 'pero', and others) due to its low importance in the teacher's evaluation context. However, it is possible to detect distinctive terms that can be considered as features (e.g. 'aburrida', 'accesible', 'mala', 'bipolar', and others).

On the features selection process, the corpus was divided into two datasets: train and test, considering the third part of the total corpus for the training, and the other two parts for the testing. After the selection process, the VIM method was executed, to calculate the terms weight of the training set. The features selection consisted on choose those higher terms weights. Table 4 shows extraction of terms (features) with its corresponding

weight through VIM method. Once the features were selected, a parameter optimization process was made with SVM and Random Forest, which is described in the next section.

**Table 4.** Extract of features obtained by VIM method

no.	features	weight	class
1	exelente	91.4665828	positive
2	buena	72.3517794	positive
3	bien	44.6535697	positive
4	demuestra	37.7831315	positive
5	atento	35.2577893	positive
6	agradable	31.4903273	positive
7	deberia	29.387665	positive
8	falta	26.2034223	positive
9	mejores	23.7956642	positive

### Parameter optimization.

SVM parameter optimization was performed using an effective method given by Hsu [25]. This method consists of an exhaustive grid search using growing values for  $C$  and  $\sigma$ . The  $C$  values used in this work for all kernels were 1, 10, 50, 80 and 100. The  $\sigma$  ( $\gamma$  in Laplacian kernel implementation) values were exponentially growing values from 0.001 to 100. The tuning procedure for the rest of Polynomial parameters was to try different values for the degree (from 2 to 10), and 0-1 for coef parameter. All these parameter values are found typical in the literature.

For each combination of parameter values, we performed 30 SVM runs by 10-fold Cross-Validation with different seeds each. We calculated the accuracy of each run and the weighted accuracy over 30 runs. We selected the combination of parameters that obtained the highest accuracy. The results of parameter optimization in the linear kernel are shown in table 5, the highest value for  $C$  is shown in bold.

The results of parameter optimization in radial basis kernel are shown in table 6. The highest value is shown in bold, and the best value for  $\sigma$  is 0.01928. The results of parameter optimization in the polynomial kernel are shown in table 7. We show only the best results obtained for each degree. The highest value is shown in bold.

**Table 5.** Results of parameter optimization in a linear kernel

C	Accuracy	Kappa	Accuracy SD	Kappa SD
0.25	0.7926	0.5846	0.058	0.1163
0.5	0.8675	0.735	0.0512	0.1023
<b>1</b>	<b>0.8804</b>	<b>0.7609</b>	<b>0.0481</b>	<b>0.0961</b>
2	0.8746	0.7493	0.0427	0.0852
4	0.8803	0.7608	0.0468	0.0934
8	0.8803	0.7608	0.0468	0.0934
16	0.8803	0.7608	0.0468	0.0934
32	0.8803	0.7608	0.0468	0.0934
64	0.8803	0.7608	0.0468	0.0934



**Table 6.** Results of parameter optimization in radial basis kernel

<b>C</b>	<b>Accuracy</b>	<b>Kappa</b>	<b>Accuracy SD</b>	<b>Kappa SD</b>
0.25	0.7926	0.5846	0.058	0.1163
0.5	0.8675	0.735	0.0512	0.1023
<b>1</b>	<b>0.8804</b>	<b>0.7609</b>	<b>0.0481</b>	<b>0.0961</b>
2	0.8746	0.7493	0.0427	0.0852
4	0.8803	0.7608	0.0468	0.0934
8	0.8803	0.7608	0.0468	0.0934
16	0.8803	0.7608	0.0468	0.0934
32	0.8803	0.7608	0.0468	0.0934
64	0.8803	0.7608	0.0468	0.0934
128	0.8803	0.7608	0.0468	0.0934

**Table 7.** Results of parameter optimization in the polynomial kernel

<b>degree</b>	<b>C</b>	<b>Accuracy</b>	<b>Kappa</b>	<b>Accuracy SD</b>	<b>Kappa SD</b>
1	0.01	0.8919	0.7841	0.0444	0.0884
1	0.08	0.8530	0.7066	0.0401	0.0799
<b>1</b>	<b>0.1</b>	<b>0.8559</b>	<b>0.7122</b>	<b>0.0360</b>	<b>0.0718</b>
1	0.5	0.8487	0.6978	0.0468	0.0932
1	1	0.8501	0.7007	0.0472	0.0941
1	1.3	0.8530	0.7064	0.0457	0.0909
2	0.01	0.8501	0.7006	0.0411	0.0819
2	0.08	0.8529	0.7063	0.0427	0.0850
2	0.1	0.8529	0.7063	0.0427	0.0850
2	0.5	0.8529	0.7063	0.0427	0.0850
2	1	0.8529	0.7063	0.0427	0.0850
2	1.3	0.8529	0.7063	0.0427	0.0850
3	0.01	0.8486	0.6977	0.0461	0.0918
3	0.08	0.8486	0.6977	0.0461	0.0918

In the case of Random Forest, this has only two tuning parameters: the number of variables in the random subset at each node and the number of trees in the forest (see table 8).

**Table 8.** Results of parameter optimization in Random Forest

<b>mtry</b>	<b>Accuracy</b>	<b>Kappa</b>	<b>Accuracy SD</b>	<b>Kappa SD</b>
77	0.8515	0.7034	0.0481	0.0957
88	0.8515	0.7035	0.0461	0.0917

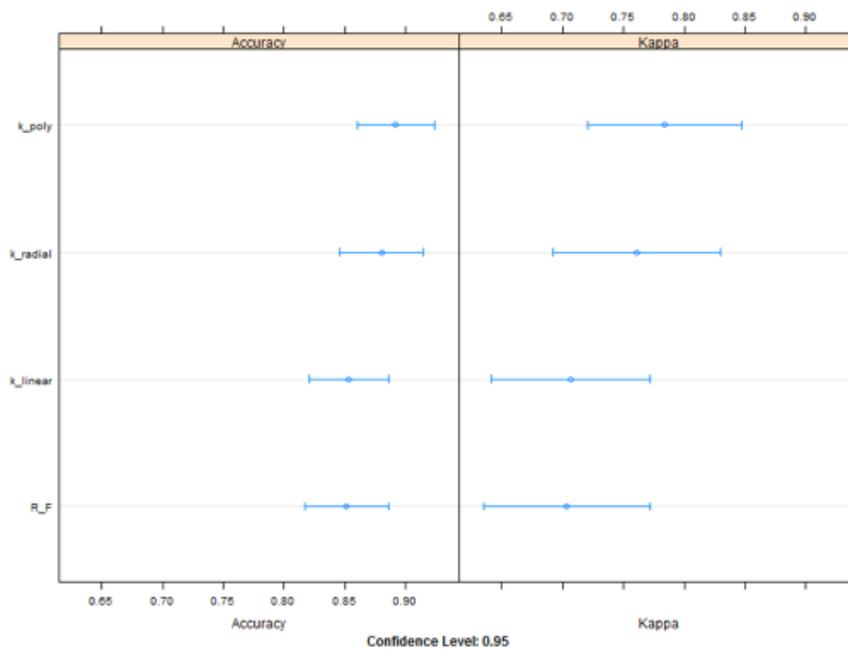
In this work, we let random forest to tune the first parameter automatically. In order to tune the second one, we performed 30 runs using different numbers of trees from 50 to 500. The highest value is shown in bold.

The average result of the optimization process for the linear kernel (k-linear), radial basis kernel (k-radial), polynomial kernel (k-poly) and Random Forest (R-F) parameters, is presented in table 9, in which can be observed kernels radial basis and polynomial presents a better performance in Accuracy and Kappa metrics.

**Table 9.** Average results in optimization process

Accuracy						
	Min.	1st Qu.	Median	Mean	3rd Qu	Max
k-linear	0.7826	0.8214	0.8696	0.853	0.8918	0.9
k-radial	0.7971	0.8514	0.8921	0.8804	0.9104	0.942
k-poly	0.8116	0.8643	0.8986	0.8919	0.9247	0.9565
R-F	0.7536	0.8194	0.8634	0.8515	0.8809	0.913
Kappa						
	Min.	1st Qu.	Median	Mean	3rd Qu	Max
k-linear	0.566	0.6429	0.7397	0.7064	0.7836	0.8
k-radial	0.5951	0.7032	0.7842	0.7609	0.8208	0.8841
k-poly	0.6245	0.7286	0.7974	0.7841	0.8494	0.9131
R-F	0.5086	0.6398	0.727	0.7034	0.7621	0.8261

Figure 8 presents the average results of the parameters optimization process (table 9) graphic.



**Figure 5.** Tuning results average of SVM and Random Forest

### Classification process

Considering the parameters obtained in the last section for SVM and Random Forest, it proceeds to measure the classification performance of the algorithm. Table 10 shows the average of the results obtained from accuracy, weighted accuracy, sensibility and specificity metrics. On the other hand, focused on sensibility metric, the SVM linear kernel overcome the radial kernel.

**Table 10.** Average of results obtained in metrics

	<b>accuracy</b>	<b>weighted accuracy</b>	<b>sensibility</b>	<b>specificity</b>
<b>k - linear</b>	0.84624277	0.8583468	0.9380117	0.75657143
	0.01345802	0.01287573	0.01593038	0.02417855
<b>k - radial</b>	0.8517341	0.86167796	0.93235867	0.77295238
	0.01485677	0.01383666	0.02250189	0.03280663
<b>k - poly</b>	0.72418113	0.80467569	0.52923977	0.97466667
	0.07601224	0.03229329	0.26285412	0.11522699
<b>R - F</b>	0.84865125	0.85777845	0.92787524	0.7712381
	0.01511305	0.01505062	0.02016065	0.024117312

The SVM polynomial kernel gets the lower performance in total accuracy, balanced accuracy and sensibility metrics, however, gets the highest performance in specificity metric. The linear and radial Kernels enhance the performance through features optimization. SVM with polynomial and Random Forest presents an improvement on Accuracy and Weighted Accuracy metrics. Despite the Polynomial Kernel decrease in sensibility, the specificity metric reaches the best performance of the others algorithms. On the other hand, Random Forest algorithm get an important decrease in specificity metric. Considering the SVM radial Kernel results in Accuracy and Weighted Accuracy, they overcome the others kernels and Random Forest, therefore a ROC analysis is made to determine through this obtained results (specificity and sensibility), the predictive values, positives or negatives (see table 11), and the ROC curve (see Figure 6).

**Table 11.** Extraction of predictive values to create ROC curve

<b>P(+)</b>	<b>P(-)</b>	<b>VP(+)</b>	<b>FV(+)</b>	<b>VP(-)</b>	<b>FVP(-)</b>
0.2341	0.7659	0.0398	0.9602	0.9991	0.0009
0.2412	0.7588	0.0773	0.9227	0.9982	0.0018
0.2482	0.7518	0.1127	0.8873	0.9973	0.0027
0.2553	0.7447	0.1461	0.8539	0.9964	0.0036
0.2623	0.7377	0.1777	0.8223	0.9954	0.0046
0.2694	0.7306	0.2077	0.7923	0.9944	0.0056
0.2764	0.7236	0.2361	0.7639	0.9935	0.0065
0.2835	0.7165	0.2631	0.7369	0.9924	0.0076
0.2905	0.7095	0.2888	0.7112	0.9914	0.0086
0.2976	0.7024	0.3133	0.6867	0.9904	0.0096
0.3046	0.6954	0.3367	0.6633	0.9893	0.0107
0.3117	0.6883	0.359	0.641	0.9882	0.0118
0.3187	0.6813	0.3803	0.6197	0.9871	0.0129
0.3258	0.6742	0.4007	0.5993	0.986	0.014
0.3328	0.6672	0.4202	0.5798	0.9848	0.0152

The table presents an extraction of results obtained in this analysis. Based on the values of table 11, Figure 6 presents the ROC curve gotten.

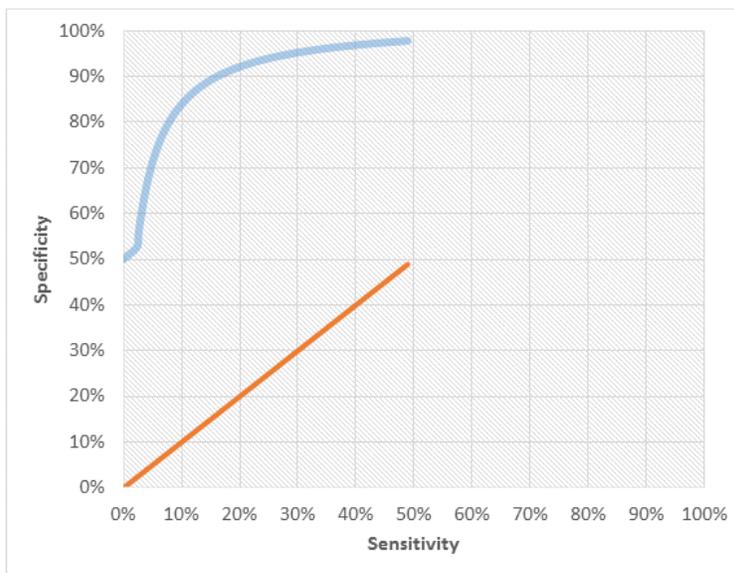


Figure 6. ROC Curve result

The obtained results of sensibility and specificity metrics are similar. Nonetheless it is important to conclude that this results obtained with machine learning algorithms, can improve even when a subjectivity corpus is not implemented. But it is such important to select features to foresee a good performance in the predictive and classification process.

## 6. CONCLUSIONS AND PERSPECTIVES

In this work, we describe the development of Sentiment Analysis Model named SocialMining, with the aim of support the teacher's performance evaluation process through the analysis of students comments, based on machine learning algorithms. Allowing contribute to improving the education of students in Mexico, one of the main motivations of this work. Despite the specific focus of teachers performance assessment, SocialMining Model is able to be adapted to different contexts as politics or e-business. This due to process contained in this model capable of analyzing sentiments in texts, which can be obtained from many ways such as social networks or databases, in addition, includes recognized algorithms by its high performance in prediction and classification process.

One of the aims of the presented study is to determine which algorithm (SVM or Random Forest) gets better performance in comments classification of teachers performance. The results obtained in this study indicate that the algorithm performance depends on the naturalness of the data used, the features selection, the parameter optimization and the use of means such as lexicons or subjectivity corpus.

In [26] a higher accuracy value is obtained, combining Naive Bayes algorithm with a subjectivity corpus. In this work, the SocialMining Model with SVM linear kernel and radial basis kernel get feasible results in accuracy and weighted accuracy metrics, as well as sensibility. However, it is highlighted again that features selection is such important to the construction of any learning model, due to its capability of ameliorate classification performance. The presence of irrelevant features generalizes low results in the classification process. Therefore, the features selection is considered ad the most important process that allows SocialMining Model flexibility and better results in comments classification process. So it is possible to conclude that an appropriate features selection consents an effective data representation, giving, as a result, a better performance

in classification and predictive process. Due to SocialMining Model is not so exhaustive on features selection, it is considered as improvement area, with the goal of enhancing the classification performance, the reason why is proposed as future work, includes a fourth phase in this model focused on features optimization considering filter methods, wrapper methods and embedded methods.

The essential part of this research is to determine which type of algorithm influences better on features optimization, considering the naturalness of the data used (text longitude, language, context, focus).

## Acknowledgement

This research was partially supported by PRODEP and CONACYT.

## REFERENCES

- [1] G. Valenzuela, Evaluacion del desempeño de los profesores. Evaluacion de Instituciones Educativas, 2008.
- [2] A. J. Shinkfield and D. L. Stufflebeam, "Teacher evaluation: Guide to effective practice," Springer Science Business Media, vol. 41, 2012.
- [3] M. C. N. Altrabsheh and S. Fallahkhair, "Learning sentiment from students' feedback for real-time interventions in classrooms," Springer International Publishing, pp. 40–49, 2014.
- [4] E. Martinez-Rodrigo and P. Raya-Gonzalez, "El microblogging en el proceso de enseñanza-aprendizaje una experiencia academica con twitter/the microblogging in the 'teaching-learning process,'" Academic experience with Twitter. Historia y comunicacion social, pp. 139, 2013.
- [5] P. Bharathisindhu and S. Brunda, "Identifying e-learner's opinion using automated sentiment analysis in e-learning," IJRET: International Journal of Research in Engineering and Technology, vol. 3, pp. 2319–2322, 2014.
- [6] B. Liu, "Handbook of natural language processing," Sentiment Analysis and Subjectivity, pp. 627–667, 2010.
- [7] H. Liu et al., "Combining user preferences and user opinions for accurate recommendation," Electronic Commerce Research and Applications, vol. 12, pp. 14–23, 2013.
- [8] B. Liu, "Sentiment analysis and opinion mining," Synthesis lectures on human language technologies, vol. 5, pp. 1–167, 2012.
- [9] J. M. A. Ortigosa and R. M. Carro, "Sentiment analysis in facebook and its application to e-learning," Computers in Human Behavior, vol. 31, pp. 527–541, 2014.
- [10] C. PongInwong and W. Rungworawut, "Teaching sentilexicon for automated sentiment polarity definition in teaching evaluation," In Semantics, Knowledge and Grids (SKG), 10th International Conference on IEEE, pp. 84–91, 2014.
- [11] N. S. P. Kaewyong, A. Sukprasert and Phang, "The possibility of students' comments automatic interpret using lexicon based sentiment analysis to teacher evaluation," The 3rd International Conference on Artificial Intelligence and Computer Science, 2015.
- [12] L. L. B. Pang and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," In Proceedings of the ACL-02 conference on Empirical methods in natural language processing, vol. 10, pp. 79–86, 2015.
- [13] M. D. S. F. Colace and L. Greco, "Safe: A sentiment analysis framework for e-learning," International Journal of Emerging Technologies in Learning, vol. 9, 2014.
- [14] A. Jebaseel and D. Kirubakaran, "M-learning sentiment analysis with data mining techniques," International Journal of Computer Science and Telecommunications, vol. 3, 2012.
- [15] V. N. Vapnik and V. Vapnik, "Statistical learning theory," International Journal of Computer Science and Telecommunications, vol. 1, 1998.
- [16] L. YungMing and L. TsungYing, "Deriving market intelligence from microblogs," Decision Support Systems, vol. 55, pp. 206–2017, 1998.
- [17] L. Breiman, "Random forests," Machine learning, vol. 45, pp. 5–32, 2001.
- [18] L. Breiman, "Bagging predictors," Machine learning, vol. 24, pp. 123–140, 1996.
- [19] W. Liu and J. Zhang, "The multilayer sentiment analysis model based on random forest," 2nd International Conference on Advances in Mechanical Engineering and Industrial Informatics, pp. 1315–1320, 2016.
- [20] A. Hjerpe, "Computing random forests variable importance measures (vim) on mixed numerical and categorical data," Master's thesis, Institute of Technology School of Computer Science and Communication, 2016.

- [21] K. A. Spackman, "Signal detection theory: Valuable tools for evaluating inductive learning," In Proceedings of the sixth international workshop on Machine learning, pp. 160–163, 1989.
- [22] F. Provost and R. Kohavi, "Guest editors' introduction: On applied research in machine learning," Machine learning, vol. 30, pp. 127–132, 1998.
- [23] U. K. S. M. Vieira and J. M. Sousa, "Cohen's kappa coefficient as a performance measure for feature selection," In Fuzzy Systems (FUZZ). IEEE International Conference on IEEE, pp. 1–8, 1998.
- [24] G. Biau and E. Scornet, "A random forest guided tour," Test., pp. 197–227, 2016.
- [25] C. W. Hsu and C. J. Lin, "A simple decomposition method for support vector machines," Machine Learning, vol. 46, pp. 291–314, 2002.
- [26] G. Gutierrez, A. Padilla, J. Canul-Reich, C. De-Luna and J. Ponce, "Proposal of a Sentiment Analysis Model in Tweets for improvement of the teaching Learning process in the Classroom using a Corpus of Subjectivity", International Journal of Combinatorial Optimization Problems and Informatics, vol. 7, no. 2, pp. 22-34, 2016.