Research Article

Umair Younis, Muhammad Zubair Asghar*, Adil Khan, Alamsher Khan, Javed Iqbal, and Nosheen Jillani

Applying Machine Learning Techniques for Performing Comparative Opinion Mining

<https://doi.org/10.1515/comp-2020-0148>

Received Jun 20, 2019; accepted Apr 16, 2020

Abstract: In recent times, comparative opinion mining applications have attracted both individuals and business organizations to compare the strengths and weakness of products. Prior works on comparative opinion mining have focused on applying a single classifier, limited comparative opinion labels, and limited dataset of product reviews, resulting in degraded performance for classifying comparative reviews. In this work, we perform multi-class comparative opinion mining by applying multiple machine learning classifiers using an increased number of comparative opinion labels (9 classes) on 4 datasets of comparative product reviews. The experimental results show that Random Forest classifier has outperformed the comparing algorithms in terms of improved accuracy, precision, recall and f-measure.

Keywords: Comparative Opinion Mining, machine learning algorithms, multi-class classification

1 Introduction

A massive volume of user-generated content is created every day on social media sites like Facebook, YouTube, and Twitter [1]. Opinion mining and sentiment analysis (SA) aims at investigating the public's opinions, sentiment, evaluations, and emotions. The Comparative Opinion Mining (COM) is a special type of opinion mining aimed at comparing the pros and cons of competing products, since

the online community relies on comparative reviews when purchasing or selling products [2]. Comparative reviews are different from the regular opinions and often comprise of a few important elements like comparative keywords, two comparable entities or compared aspects [3]. Apart from directly expressing positive or negative opinions about a product or an entity and its aspects, one can also express the opinions by comparing the two entities- such opinions are called *comparative opinions* [4].

Comparative opinion mining assists business organizations to analyze customer feedback about their manufactured goods in terms of both strengths and weaknesses. For example, the input review: "Android is better IOS is too expensive", is a comparative review comparing two operating systems, namely Android and IOS. Collection and analysis of such comparative opinions is beneficial for business organizations to enhance the quality of their manufactured goods and individuals when buying products. Therefore, it is necessary to develop comparative opinion mining systems [5].

The early studies on comparative opinion mining have used limited datasets (400 reviews dataset) with a limited number of machine learning classifiers [6–8]. Hence, there is a need to explore multiple machine learning classifiers with datasets of increased sizes for performing comparative opinion mining based on comparative reviews for different products. Therefore, this study aims at conducting and applying different machine learning classifiers for performing comparative opinion mining and recommending the classifier with best results.

The baseline study [9] used a single Naïve Bayes classifier with small scale dataset (400 reviews dataset). The major limitation of their work is the use of a single classifier on a dataset of small size (400 reviews), which yielded poor performance. The proposed work is an enhancement of the work performed by [9] by using multiple classifiers, experimented on 4 extended datasets (6000, 3000, 3000, 1000 respectively). Moreover, we used an increased number of Joint Label Reviews, divided into 9 classes. This work aims at performing comparative opinion mining using supervised machine learning techniques by enhancing the

*Corresponding Author: **Muhammad Zubair Asghar:** Institute of Computing and Information Technology, Gomal University, D.I. Khan (KP), Pakistan; Email: zubair@gu.edu.pk

Umair Younis, Alamsher Khan, Javed Iqbal, Nosheen Jillani: Institute of Computing and Information Technology, Gomal University, D.I. Khan (KP), Pakistan

Adil Khan: Department of Computer Science, University of Peshawar, Peshawar, KP, Pakistan; School of Computer Science, SZIC, University of Peshawar, Peshawar, KP, Pakistan

work performed by Khan *et al.* (2016). What follows is a list of research questions to be addressed in this study: RQ1: How to perform a comparison of different opinion mining classifiers on different datasets using supervised machine learning classifiers? RQ2: What is the efficiency of the different machine learning techniques for performing a comparison of different opinion mining classifiers for online user reviews? RQ3: Which one is the most efficient opinion mining classifier based on the comparison of different machine learning classifiers?

Research Contributions

To perform comparative opinion mining with respect to applying supervised machine learning algorithms on different product review datasets, we selected a baseline study performed by [9] and the proposed work is significant for the following reasons: (1) Applying multiple machine learning algorithms on product review datasets for performing comparative opinion mining, which will assist opinion mining researchers design more complex applications for comparative opinion mining, and (2) the performance evaluation results will assist in choosing the classifier in best classification results for comparative opinion mining, and (3) Development of four new datasets for conducting comparative opinion mining.

2 Related Work

In this section, we presented some of the selected studies for conducting comparative opinion mining.

Research community studied the phenomenon of comparative opinion mining using different machine learning approaches such as supervised machine learning and hybrid machine learning techniques. For example, to perform comparative opinion mining, a supervised machine learning technique, based on Naïve Bayes classifier, is proposed by [9]. The dataset is acquired from YouTube about two products, namely iOS and Android. The system is based on multi-label classification. The major limitation of their work is the use of a single classifier on a dataset of small size, which yielded poor performance. In other notable work, to conduct contrastive opinion summarization, [10] proposed a supervised machine learning approach using blog threads in product reviews. Improved results are achieved when compared with the baseline methods. The work has a limited dataset size with a limited number of machine learning classifiers. However, further, improvement can be made by incorporating a feature set [10].

Some work focuses on product domain to perform comparative opinion mining. As an example, in their work on Vietnamese product reviews, [6] proposed an opinion mining system based on comparative reviews in the product domain. They applied different supervised machine learning techniques and obtained promising results. However, further, improvement can be made investigating other machine learning classifiers with a revised feature set [6]. [7] proposed a model of comparative analysis of classification algorithms wherein three machine learning algorithms, namely N.B, Random Forest and K* were applied. Experiments were conducted on three datasets and satisfactory results obtained. The work can be extended by using more datasets. [6, 7] need to investigate other machine learning classifiers with a revised feature set. Further, a fusion of two supervised machine learning techniques and three greedy algorithms for comparative opinion mining is proposed by [20]. They obtained satisfactory results by finding the optimal solution. Further improvement is possible by presenting visual summary reports for enhanced customer support. In other literature, comparative opinion mining is also done using a movie review dataset as proposed by [11], wherein they proposed a system for sentiment classification of movie reviews using different Machine Learning Algorithms, namely Naïve Bayes, Linear support vector machine, and Synthetic word approach. A benchmark dataset, namely the Internet Movies Database (IMDB), was used to conduct experiments. The experimental results show that, overall, Linear SVM provides the best accuracy followed by Synthetic word Approaches. However, the system can experiment with multiple datasets in different domains. Another related study by [12] evaluated the performance of different machine learning classifiers with respect to sentiment prediction on the Twitter dataset (movie reviews). Results show that SVM obtained better results over other classifiers. However, the performance of the system is yet to be verified for multilingual support with an extended set of performance evaluation matrices [13]. Following the line of exploiting Twitter reviews for comparative opinion mining, Ismail *et al.* (2016) proposed a system of comparative analysis for Twitter sentiment analysis, with a special focus on Twitter reviews. Most commonly used supervised machine learning classifiers are explored. Overall, Naïve Bayes outperformed the other classifiers. The work can be extended by conducting experiments on a larger representative dataset. Furthermore, incorporating semantic features and evaluation of classification algorithms can enhance the system's performance. Different supervised learning approaches such as SVM, NB and Maximum Entropy, were evaluated to predict an optimal classifier for performing SA [13]. The exper-

Table 1: A Review of Selected Studies

Study	Aim of the Work	Technique(s)	Datasets	Performance	Limitations	Future work
Varathan <i>et al.</i> [2]	COM.	Machine Learning: (SVM) Unsupervised: (Clustering) Maximum Entropy (MEM) SVM	Manually compiled dataset, from Amazon. Dataset is collected from Tinhte.com and VnReview	SVM: 79.8%	Only MLP based approaches are used.	Experimentation other products review datasets
Bach <i>et al.</i> [6]	Mining comparative sentences in Vietnamese			SVM: 86.30% MEM: 73.48%	Targets only one domain (electronics)	System can be extended by explore the models for comparative reviews.
Duriqi <i>et al.</i> [7]	M.L system uses the most popular classifiers	. N.B . Random Forest . K* Algorithm	Datasets used are: . Diabetes . Spam Base . Credit Approval Amazon products reviews. (tech products)	Diabetes: 76.3 S.B: 79.28 C.A: 77.68	Accuracies are not up to the mark in w.r.t to all three datasets.	Using more datasets, also with the usage of other classifiers
Ejaz <i>et al.</i> [8]	SVM, NB	Lexicon-based approach used with n-grams and ML classifiers Naïve Bayes	User reviews on YouTube comprising of iPhone and Android videos. Movie Review dataset	SVM: 71% N.B: 73%	Failed to predict reviews contains ambiguities	Assigning different weights to reviews.
Khan <i>et al.</i> [9]	COM			NB: 33%.	Performance is not up to the mark.	By using more ML classifiers to enhance the performance.
Vidhita <i>et al.</i> [10]	COS	PLSA + COS		Accuracy: 78.9% Precision: 0.708 Recall: 0.891 F-Measure: 0.9	Small dataset (Limited no. of reviews)	Larger data set can be
Rana & Singh [11]	Comparative analysis of user reviews	Naïve Bayes SVM, Rapid Minor	Internet Movie Database	SVM provides best accuracy (82 %).	Limited dataset. (movies and dramas)	Explore data of a wider range of various items/products.
Joshi & Tekchandani [12]	Supervised M.L classifiers	Machine Learning Classifiers. SVM, Naïve Bayes and Maximum Entropy	Dataset is extracted from Twitter about Movie Reviews	SVM gives better result with accuracy of 84%.	Only dataset of movie reviews are used.	System can be enlarged automated sentiments classifier for more languages.
M D <i>et al.</i> [13]	Comparative study on diverse methodologies	Machine Learning Techniques (SVM, NB, Maximum Entropy K-NN) . Naïve Bayes . Support Vector Machine . Random Forest NB, SVM)	Social Network Blogs (FB, Twitter) Twitter feed of standard movie reviews. Dataset has been used in Non-English Languages.	Accuracy of overall results up to 75%. N.B = 89% SVM = 88% Random Forest = 85% NB: 76.6 SVM: 78.4	Selective methods were used for comparative SA	By adopting some more methods or ML techniques.
Nayak and Nataranjan [14]	Comparative study of different M.L classifiers				There is no definite ranking, which classifier works best.	Using other machine learning techniques.
Madagoda <i>et al.</i> [15]	OM sentiment classification				Only based on the statistical methods, only polarity is determined	Adopting more efficient classifiers.
Ganapathibhotla & Liu [16]	Sentiment Analysis	Bootstrapping based on word net.	Dataset is collected from (Amazon)	Precision: 0.843 Recall: 0.857 F-Measure: 0.873	Need a measure of association of comparative word.	Adopting new method called OSA.
Ghag & Shah [17]	Comparative analysis of	SVM, N.B, Random Forest	Self-designed training set is used	SVM: 74.5 NB: 70.0 Random Forest: 77.45	Lack of support for multilingual SA.	Work can be extended by generalized SA without using trained dataset.
Bhavitha <i>et al.</i> [18]	S.A for comparative reviews.	Supervised ML	Movie reviews, Tech Product reviews	SVM: 85%	Need of automatic analysis	More methods of Classification for Comparative Sentences.
Lerman & McDonald [19]	COS on customer reviews	SAM + COS	Consumer electronics	SAM + Contrastive = 1.68	Roughly have average scores.	Scores may further be enhanced as a future work.

iments were conducted on a dataset of social blogs (Facebook and Twitter) and satisfactory results are achieved. More advanced techniques of SA (rule base, lexicon base, and other ML classifiers) can be investigated for better results. The research study by [14] conducted a comparative study on machine learning classifiers for sentiment analysis of Twitters feeds. Three prominent supervised machine learning classifiers are employed on a Twitter feed dataset. The work can be extended in the future by using other machine learning classifiers like Latent Sentiment Analysis and other techniques.

From the perspective of hybrid machine learning approaches, a lot of research work has been performed by scientists. For instance, a comparative opinion mining framework is proposed by [8] by n-gram features, lexicons and a set of machine learning algorithms. An Amazon-based dataset is used to conduct experiments and results are compared with baseline methods. The study faces a drawback of exploiting a limited dataset with a limited number of machine learning classifiers. However, further, improvement can be made by assigning appropriate weights to features based on product popularity. In the context of hybrid machine learning approaches another similar work on a comparative opinion mining system is proposed by [2], using a technique that was a merger of clustering-based (unsupervised) and Support Vector Machine (supervised) methods. Promising results are obtained w.r.t comparing methods. However, more robust results can be obtained by performing experiments in multiple domains with additional preprocessing steps. [15] performed a comparative analysis of user reviews in non-English languages using supervised and unsupervised classifiers. Experimental results show the effectiveness of applied methods. However, more efficient results can be achieved by applying the techniques on larger datasets in different domains.

Some existing studies in the area of comparative opinion mining have exploited variant approaches: [16] proposed a comparative opinion mining system using different types of sentences such as comparative and superlative. For this purpose, different datasets were acquired from Amazon, CNet and Howard forums. In addition, [17] conducted a comparative study on the performance evaluation of various techniques used in sentiment analysis. A self-designed training set is used for the comparison. The results obtained are random, showing the effectiveness of different works on different datasets. The main challenge identified is to obtain results, being extracted using a generalized sentiment analysis system having no dependency on a training dataset. Similarly, [18] performed a comparative analysis of different ML classifiers for sentiment analysis. The input text is analyzed in different dimensions,

such as identification of subjective text, expressions and attitude hidden inside the text. Different datasets like comparative products and movie reviews are used to conduct the experiments. Satisfactory results are obtained. Some research studies focus on the opinion summarizer such as [19] proposed a contrastive opinion summarizer model to analyze summaries and highlight their differences. Their proposed method is based on the standard additive model (SAM) and contrastive opinion summarization (COS) and satisfactory results are obtained. Incorporation of the aggregated scoring mechanism can enhance the capability of the proposed method.

The aforementioned works on comparative opinion mining using online social media based user generated content have exploited variant techniques, like supervised (Support Vector Machine, Naïve Bayes, Maximum Entropy, and Random Forest), and hybrid approaches (supervised, unsupervised, and lexicon based approach) on a small scale dataset. However, it is necessary to examine the implementation of multiple classifiers on extended datasets for the efficient classification of comparative opinion mining. Table 1 presents A Review of Selected Studies.

3 Method

The proposed work consists of the following tasks: (1) Data Collection and Annotation, (2) Applying Pre-processing techniques, (3) Applying machine learning classifiers, (4) Performance evaluation of different classifiers, and (5) recommending classifier with improved performance for comparative opinion mining.

3.1 Dataset Collection & Annotation

The dataset compilation module is used to compile datasets from comparative user reviews, which serve as an input to the noise reduction module for filtering the noisy text. For this purpose, we acquired 3 datasets of comparative reviews, namely iOS vs Android, Facebook vs Twitter, Microsoft vs Google and Pearl Continental vs Marriott. A detailed description of the three datasets is presented in Table 2. The data is collected from YouTube, which is publicly available data and there is no personally identifiable information of the forum users that is collected or used for this study. For this purpose, beautiful soup (Leonard Richardson, 2012) is used to scrap the required reviews. The four datasets comprised of 6000, 3000, 3000 and 1000 comparative reviews respectively.

Table 2: Details of Acquired Datasets

D#	Description	No. of Reviews	No. of Reviews w.r.t classes								
			<i>pos_pos</i>	<i>pos_neg</i>	<i>pos_neu</i>	<i>neg_neg</i>	<i>neg_pos</i>	<i>neg_neu</i>	<i>neu_neu</i>	<i>neu_pos</i>	<i>neu_neg</i>
D1	iOS vs Android	6000	826 (13.76%)	2753 (45.88%)	611 (10.18%)	110 (1.83%)	739 (12.31%)	64 (1.06%)	544 (9.06%)	310 (5.16%)	49 (0.81%)
D2	Facebook vs Twitter	3000	440 (14.66%)	1208 (40.26%)	444 (14.8%)	55 (1.83%)	307 (10.23%)	59 (1.96%)	311 (10.36%)	143 (4.76%)	31 (1.03%)
D3	Microsoft vs Google	3000	360 (12%)	1263 (42.1%)	401 (13.36%)	62 (2.06%)	379 (12.63%)	46 (1.53%)	321 (10.7%)	149 (4.96%)	30 (1%)
D4	Pearl continental vs Marriott	1000	225 (22.5%)	275 (27.5%)	55 (5.5%)	50 (5%)	145 (14.5%)	45 (4.5%)	55 (5.5%)	155 (15.5%)	50 (5%)

The sentences are annotated into positive, negative and neutral classes. The three classes are sub-divided into 9 classes (*pos_neg*, *pos_pos*, *pos_neu*, *neg_pos*, *neg_neg*, *neg_neu*, *neu_neu*, *neu_neg*, *neu_pos*). The data sets are made available at: <https://www.kaggle.com/umairyounis/comparative-reviews-datasets>

The acquired data sets are split into training (80%) and testing (20%) for experimentation.

A detailed distribution of the different classes is presented in the Table 2, as follows:

The Dataset-1, namely “iOS vs Android” comprised of 6000 reviews, in which different classes have the following number of reviews: *pos_neg* = 2753, *pos_pos* = 826, *pos_neu* = 611, *neg_pos* = 739, *neg_neg* = 110, *neg_neu* = 64, *neu_neu* = 544, *neu_neg* = 49, and *neu_pos* = 310. The Dataset-2, namely “Facebook vs Twitter” comprised of 3000 reviews, in which different classes have the following number reviews: *pos_neg* = 1208, *pos_pos* = 440, *pos_neu* = 444, *neg_pos* = 307, *neg_neg* = 55, *neg_neu* = 311, *neu_neu* = 544, *neu_neg* = 31, and *neu_pos* = 143. The Dataset-3, namely “Microsoft vs Google” comprised of 3000 reviews, in which different classes have the following number reviews: *pos_neg* = 1263, *pos_pos* = 360, *pos_neu* = 401, *neg_pos* = 379, *neg_neg* = 62, *neg_neu* = 46, *neu_neu* = 321,

neu_neg = 30, and *neu_pos* = 149. The Dataset-4, namely “Pearl Continental vs Marriott” is comprised of 1000 reviews, in which different classes have the following number reviews: *pos_neg* = 275, *pos_pos* = 225, *pos_neu* = 55, *neg_pos* = 145, *neg_neg* = 50, *neg_neu* = 45, *neu_neu* = 55, *neu_neg* = 50, and *neu_pos* = 155.

Training Data

Three human annotators were asked to assign polarity classes manually to reviews in the training dataset. The manual annotation performed by three annotators yielded three votes for each review and the review was assigned polarity class using majority voting. An inter-annotator agreement {83.4%}, having a Kappa (K) score of 0.84 is achieved, which is quite satisfactory. The Kappa score is a frequently used computation for validating the inter-annotator agreement [21], showing the degree of agreement that exists among human annotators. The training dataset is comprised of 80% (D1: 4800, D2: 2400 and D3: 2400) reviews. The sample of training dataset is shown in Table 3.

Table 3: Set of Reviews from the Dataset (Training)

Review No.	Review	Polarity Label
1.	i own android and iOS both are fantastic operating systems	<i>pos_pos</i>
2.	android is better than iOS	<i>pos_neg</i>
3.	iOS is hit android is just okay	<i>pos_neu</i>
4.	android and iOS both sucks	<i>neg_neg</i>
5.	iOS would be crushed by great android	<i>neg_pos</i>
6.	iOS has no ability to easy file sharing with android	<i>neg_neu</i>
7.	i like android but also likes iOS	<i>neu_neu</i>
8.	i have used both android and iOS but i prefer android	<i>neu_pos</i>
9.	android is fine iOS is rough	<i>neu_neg</i>

Table 4: Set of Reviews from the Dataset (Testing)

Review No.	Review	Polarity Label
1.	iOS holds the quality android dont have quality	pos_neg
2.	iOS has a smaller number of apps than android	neg_pos
3.	i am sure android and iOS will catch up with features	neu_neu

Table 5: Reviews before and after cleansing reviews.

Review #	Before Cleansing	After Cleansing	
		Stop Words Removed	Tokenized
1.	Android and iOS, both sucks	Android iOS sucks	Android (Adj) iOS (Noun) sucks (Verb)
2.	I own Android and iOS, both are fantastic operating systems	I Android iOS fantastic operating systems	I (Pronoun) android (Adj) iOS (Noun) fantastic (Adj) operating system (Noun)
3.	iOS has less number of apps than Android	iOS less number apps Android	iOS (Verb) less (Adj) number (Noun) apps (Noun) Android (Adj)
4.	Android is better than iOS	Android better iOS	Android (Noun) better (Adj) iOS (Noun)
5.	iOS would be crushed by great Android.	iOS crushed great Android	iOS (Noun) crushed (Verb) great (Adj) Android (Noun)\y

Testing Data

A test set is also called as a test dataset or testing data. It works independently of the training set. The evaluation of the model is carried out by providing test data, which acts as a gold standard. The test data is provided when the model gets completely trained. The test data is usually made available during classifier testing. It is used for the assessment of the classifier either it is working properly or not. Train, Validation and Test set [22].

We used random split method to split the dataset into training and testing. In Random Splitting, the dataset is divided into train and test in a predefined ratio. For example, in 80:20 split, we take the samples randomly. The random split method is more robust as compared to other methods, as the dataset is partitioned more accurately [23]. In this work, we randomly split the dataset into training testing in the ratio of 80:20.

The sample of the testing dataset is shown in Table 4.

We stored datasets in MS Excel files and then converted into CSV files for experimental setup. The sample of reviews for the training and test sets are shown in Table 3 and Table 4 respectively. In this work, we presented comparative reviews datasets gathered from YouTube. We also presented our method for collecting and annotating the datasets in Data collection and annotation section. The training and testing data is cleansed and processed shown in Table 5 before it is passed through machine learning classifiers [24].

3.2 Applying Preprocessing Steps

Different pre-processing techniques are applied to the acquired datasets to reduce noise for efficient classification. These techniques include tokenization, part of speech tagging, and removal of other special characters [25]. Table 5 shows text with and without preprocessing.

Tokenization

Tokenization is the process in which words are segmented into the small pieces of text. For this purpose, Python-based NLTK tokenizer is used.

Stop Word Elimination

Stop words do not show any opinion. A python-based script is implemented to eliminate such words using a predefined list. For example, “the”, “is”, “an” etc. are the examples of stop words.

3.3 Applying Machine Learning Classifiers

We applied different supervised machine learning technique such as Naïve Bayes, Logistic Regression, Support Vector Machine (SVM), KNN (K-Nearest Neighbour), Decision Tree, Random Forest, Gradient Boosting. For this purpose, the datasets are partitioned into train and test splits

Figure 1: Supervised Learning Technique for Comparative Opinion Mining

and the reviews are labeled into multiple classes (pos_neg, pos_pos, pos_neu, neg_pos, neg_neg, neg_neu, neu_neu, neu_neg, neu_pos). Figure 1 shows a generic diagram of a supervised machine learning technique.

3.3.1 Feature Engineering

Following feature selection steps are applied during the implementation of machine learning algorithms: (1) Feature Vector (2) Term Frequency (tf) (3) tf and idf

Feature Vector: It transforms a tweet into a matrix of token counts [26].

Term Frequency (tf): It counts a number of occurrences of a term in given tweets [26].

Term Frequency (tf) and Inverse Document Frequency (idf): The tf and idf computation shows the importance of a word in a given document or dataset [26].

In the following Sub-sections, the different M.L Classifiers, used in this work are summarized.

3.3.2 Naïve Bayes

Naïve Bayes classifiers belong to a set of probabilistic classifiers, relying on the main applications of the Bayes theorem. Naïve Bayes is a supervised machine learning algorithm, used for both regression and classification. The attributes and features are pretended to be tentatively independent of each other, which form a naïve assumption. The Naïve Bayes (NB) is a simple classifier to be applied and good results are acquired in most of the cases. It is easily applicable on the larger datasets. Naïve Bayes (NB) adopts all the features that are in the dataset, are equally independent and important [14]. The mathematical formulation is given as under.

$$P(XY) = \frac{P(xy)P(x)}{P(y)} \quad (1)$$

The above equation represents the following: (1) $P(x|y)$ is the probability of the class (x , target) given predictor (y , attributes), (2) $P(x)$ is the prior probability of the class, (3)

$P(y)$ is the prior probability of the *response (target) variable*.

3.3.3 Random Forest

Random Forest (RF) is the most flexible and easy-to-use machine learning classifier based on hyper parameters tuning. It produces efficient results most of the times. It is used for both regression and classification tasks. Random Forest (RF) is an ensemble-based machine learning algorithm. The outcomes of the Random Forest are the focused results of each decision tree. The higher the number of Decision Trees in the Random Forest, the more improved the generalization is [14]. The mathematical formulation is given as under.

$$y = \frac{1}{Z} \sum_{z=1}^x yz(x') \quad (2)$$

The above equation represents as follows: (1) number of the sample with the replacement, z training examples from the x , y , and (2) Training a regression or classification tree yz .

3.3.4 Support Vector Machine

Support Vector Machine is a supervised machine learning algorithm used for classification (see Algorithm ??). The working of SVM works by sorting the data into diverse classes by discovering a line which is also called hyperplane which split up the data set into classes.

The fundamental idea underlying Linear SVM for the sentiment classification is to determine a hyperplane, which splits the corpus or documents [14]. The mathematical formulation is given as under.

$$D = \{(x_1, y_1), (x_2, x_2), \dots, (x_n, y_n)\}. \quad (3)$$

The explanation of the above equation represents as follows: D = Dataset, x will be related to a value of y representing if the element belongs to the class.

3.3.5 Logistic Regression

Logistic Regression (LR) targets at classifying the reviews into different polarity classes based on training and testing datasets. It predicts to which polarity class, the review belongs. The Logistic Regression (LR) is considered as the fastest predicting classifier, makes model less difficult and

which can permit it to better generalization i.e. to avoid overfitting, and performs very well on the new data [38]. The mathematical formulation is given as under.

$$e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2} \quad (4)$$

The above equation is explained as follows: e denotes the constant, while the remaining equation represents the boundary functions.

3.3.6 K-Nearest Neighbour

The KNN algorithm is based on instance-oriented learning, used for solving classification and regression problems. It is a lazy learner, storing all instances of training dataset in the n -dimensional slot. It classifies new cases through a majority vote of its k neighbors [38]. The mathematical formulation is given as under.

$$p = \frac{1}{N} \sum_{i=1}^N a_i \quad (5)$$

The above equation represents the following: Wherever a_i is the i th case of the samples and p is the predicted result.

3.3.7 Gradient Boosting

This technique is used to make a prediction out of large data with efficient performance. It is an ensemble of machine learning techniques by combining the prediction of different estimators for improving the performance with respect to uni-estimators [38]. The mathematical formulation is given as under.

$$f(a) = z - F_n(a) \quad (6)$$

The above equation is explained as: $f(a)$ is a function that is to be minimized, here F is a positive constant, n is a total number of points. z denotes to the function.

3.3.8 Decision Tree

It is a supervised learning technique, frequently used in text classification and other problems. In this technique, instances are classified on the basis of sorting of feature values. The feature is represented as a node in the decision tree, which is to be classified. Meanwhile, the branch represents the value of a feature. Classification starts at the root node by sorting the feature values. The tree construction takes place using a divide and conquers approach [38].

The mathematical formulation is given below.

$$(a, b) = (x_1, x_2, x_3 \dots x_{k,b}) \quad (7)$$

The above equation is explained as follows:

- a denotes the root node, b is a subset, and x denotes the leaves in a tree.

3.4 How Proposed System works for Comparative Reviews Classification

The supervised learning based comparative opinion mining system works by taking input reviews, applying pre-processing steps, and classifying it into different polarity classes, namely: pos_neg, pos_pos, pos_neu, neg_pos, neg_neg, neg_neu, neu_neu, neu_neg, and neu_pos.

Dataset is split into training (80%) and testing (20%). During the training phase, the classifier is given with both tag “polarity label” and review. Once the training of classifier is completed, the efficiency of the machine learning classifiers is evaluated using test data and results are recorded in terms of different performance evaluation measures, namely accuracy, precision, recall, and f-score.

Algorithm 1 shows the Pseudocode of the proposed system

4 Results and Discussions

In this chapter, results are summarized by answering the posed research questions.

Algorithm 1 Pseudocode of the Proposed System

```

Input:                set of reviews in Comparative datasets stored in excel sheet (.CSV)
Output:               Classified Reviews w.r.t Polarity
Polarity Classes:     ["Positive", "Negative", "Neutral"]
Polarity Sub-classes: [pos_neg, pos_pos, pos_neu, neg_pos, neg_neg, neg_neu, neu_neu, neu_neg, neu_pos]
Classifiers:         ["SVM", "NB", "KNN", "Decision Tree", "Random Forest", "Logistic Regression", "Gradient Boosting"]
Stop words:          [is, a, by,.....]
Begin

//Reading
Text =                Read Text

#Pre-Processing (Tokenization/Stop Words Removal/Punctuations)

#Tokenization
Token =               Tokens (Text)

# StopWords Removal
Ptext = Remove_Stop Words (Tokens)

# Splitting Dataset into Training/Testing
X_Train, Y_Train, X_Test, Y_Test = Split (ptext, test_size = 20%)

# Creating Count_Vector (ptext)

#Tf-IDF
#Applying Machine Learning Classifier
Model =               classifier ()
Classification = Model: fit(X_Train, Y_Train)
#Prediction
Prediction =          Classification: Prediction (x-text)
#Accuracy
Accuracy =            Accuracy (Prediction, t_text)
#Confusion Matrix
CF =                  Confusion Matrix (Y_Test, Prediction)
#Performance          Measure (Precision, F_measure)
Output =              classification_report (Y_Test, Prediction, Polarity_Class)
Return (output)

```

4.1 Answer to RQ. 1

To answer RQ.1 “How to perform comparison of different opinion mining classifiers on different datasets using supervised machine learning classifiers?”, we applied different machine learning classifiers on different datasets for performing comparative opinion mining and to perform a comparison of different classifiers. To compare the performance of different classifiers for performing comparative opinion mining on multiple datasets, we used different performance evaluation measures, namely: (1) accuracy, (2) precision, (3) recall, and (4) F1-score.

Algorithm 2 shows the steps of the performance evaluation of the different machine learning classifiers

After performing the evaluation of different machine learning algorithms in terms of comparative opinion mining, we recommend Random Forest (RF) classifier with best results for comparative opinion mining.

4.2 Answer to RQ.2

To answer RQ.2 “What is the efficiency of different machine learning techniques for performing comparison of different opinion mining classifiers for online user reviews?”, we applied different machine learning classifiers on different datasets. Detail of each experiment is given as follows:

Algorithm 2 Pseudocode of the Performance Evaluation Measures

```
#Performance:
SET TC to COUNT (Prediction =Y_Test)
SET ACCURACY to TC/N2
SET TP to COUNT (Prediction = POS AND Y_Test = POS)
SET TN to COUNT (Prediction = NEG AND Y_Test = NEG)
SET FP to COUNT (Prediction = POS AND Y_Test = NEG)
Set FN to COUNT (Prediction = NEG AND Y_Test = POS)
SET PRECISION to TP/(TP+FP)
SET RECALL to IP/(TP+FN)
SET CFM to {}
CFM['TP'] = TP
CFM['FN'] = FN
CFM['FP'] = FPCFM['TN'] = TN
```

Experiment#1:

This experiment is conducted on the dataset 1, comprising of 6000 reviews about “iOS vs Android”. Table 6 shows the experimental results obtained by applying different machine learning classifiers, namely NB, LR, SVM, KNN, DT, RF, and GB, in terms of different evaluation measures like accuracy, precision, recall, and f1-score. It is evident that *Random Forest* outperformed other classifiers in terms all of the applied evaluation metrics, by achieving promising results in terms of accuracy (80%), precision (80%), recall (80%), and F-score(80%). The compile time of the all machine learning classifiers has been reported as shown in Table 6.

Table 6: Comparative results of different machine learning classifiers on dataset#1

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Compile time (ms)
Naïve Bayes	62	67	63	55	53.8
Logistic Regression	69	68	69	66	56
Support Vector Machine	78	79	79	78	418
K-Nearest Neighbour	61	59	60	61	518
Decision Tree	77	77	77	77	57.8
Random Forest	80	80	80	80	67.6
Gradient Boosting	77	70	68	65	237

Table 7: Comparative results of different machine learning classifiers on dataset#2

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F-Score (%)
Naïve Bayes	70	74	71	65
Logistic Regression	79	79	79	76
Support Vector Machine	93	94	94	94
K-Nearest Neighbour	59	59	59	57
Decision Tree	94	95	94	94
Random Forest	95	96	95	95
Gradient Boosting	86	88	87	86

Experiment#2:

This experiment is conducted on the dataset 2, comprising of 3000 reviews about “Facebook vs Twitter”. Table 7 shows the experimental results obtained by applying different machine learning classifiers, namely NB, LR, SVM, KNN, DT, RF, and GB, in terms of different evaluation measures like accuracy, precision, recall, and f1-score. It is evident that *Random Forest* outperformed other classifiers in terms all of the applied evaluation metrics, by achieving promising results in terms of accuracy (95%), precision (96%), recall (95%), and F-score (95%).

Experiment#3:

This experiment is conducted on the dataset 2, comprising of 3000 reviews about “Microsoft vs Google”. Table 8 shows the experimental results obtained by applying different machine learning classifiers, namely NB, LR, SVM, KNN, DT, RF, and GB, in terms of different evaluation measures like accuracy, precision, recall, and f1-score. It is evident that *Random Forest* outperformed other classifiers in terms all of the applied evaluation metrics, by achieving promising results in terms of accuracy (93%), precision (93%), recall (93%), and F-score (93%).

The domain of the three datasets used in the study is the same, i.e. “technological products” such as iOS vs Android, Microsoft vs Google, and Facebook vs Twitter. Therefore, due to the same domain of the datasets used, the ro-

bustness of the classifier is questionable, that’s why further experimentations are performed on the datasets of new domain “Hotel Reviews Dataset namely Pearl Continental vs Marriott”, good results are achieved on the new dataset.

Experiment#4:

This experiment is conducted on the dataset 4, comprising of 1000 reviews about “Pearl continental vs Marriott”. Table 9 shows the experimental results obtained by applying different machine learning classifiers, namely NB, LR, SVM, KNN, DT, RF, and GB, in terms of different evaluation measures like accuracy, precision, recall, and f1-score. It is evident that *Random Forest* outperformed other classifiers in terms all of the applied evaluation metrics, by achieving promising results in terms of accuracy (100%), precision (100%), recall (99%), and F-score (99%).

Experiment#5:

We applied 10-Fold Cross Validation for performing experiments on different classifiers. Result reported in Table 10 depict values of Mean of Accuracy, Standard Validation of Accuracy.

Table 8: Comparative results of different machine learning classifiers on dataset#3

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F-Score (%)
Naïve Bayes	71	74	71	66
Logistic Regression	76	77	77	75
Support Vector Machine	85	86	85	85
K-Nearest Neighbour	61	60	61	59
Decision Tree	93	94	94	93
Random Forest	93	93	93	93
Gradient Boosting	84	86	84	84

Table 9: Comparative results of different machine learning classifiers on dataset#4

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F-Score (%)
Naïve Bayes	78	77	74	60
Logistic Regression	77	78	77	77
Support Vector Machine	89	93	90	90
K-Nearest Neighbour	69	68	67	67
Decision Tree	85	89	89	89
Random Forest	100	100	100	99
Gradient Boosting	80	83	80	80

Table 10: Results by Using Cross Validation

Classifiers	Dataset #1 (iOS vs Android)	Dataset #2 (Facebook vs Twitter)	Dataset #3 (Microsoft vs Google)	Dataset #4 (Pearl Continental vs Marriott)	Mean Accuracy (Avg)	Standard Deviation (Avg)
Naïve Bayes	68	75	77	80	0.755	0.077
Logistic Regression	70	80	78	79	0.776	0.064
Support Vector Machine	89	95	90	91	0.913	0.082
K-Nearest Neighbour	65	63	69	70	0.662	0.059
Decision Tree	80	95	97	93	0.912	0.079
Random Forest	90	98	95	99	0.955	0.087
Gradient Boosting	79	88	87	82	0.844	0.080

Table 11: Performance Evaluation Results of different Machine Learning Classifiers (D1 = Dataset 1, D2 = Dataset 2, D3 = Dataset 3, D4 = Dataset 4)

Classifier	Accuracy	Precision	Recall	F-1
Naïve Bayes	D1 = 62	D1 = 67	D1 = 63	D1 = 55
	D2 = 70	D2 = 74	D2 = 71	D2 = 65
	D3 = 71	D3 = 74	D3 = 71	D3 = 66
	D4 = 78	D4 = 77	D4 = 74	D4 = 70
	Avg = 70.25	Avg = 73	Avg = 69.75	Avg = 64
Logistic Regression	D1 = 69	D1 = 68	D1 = 69	D1 = 66
	D2 = 79	D2 = 79	D2 = 79	D2 = 76
	D3 = 76	D3 = 77	D3 = 77	D3 = 75
	D4 = 77	D4 = 78	D4 = 77	D4 = 77
	Avg = 75.25	Avg = 75.25	Avg = 75	Avg = 72.33
Support Vector Machine	D1 = 78	D1 = 79	D1 = 79	D1 = 78
	D2 = 93	D2 = 94	D2 = 94	D2 = 85
	D3 = 85	D3 = 96	D3 = 85	D3 = 85
	D4 = 89	D4 = 93	D4 = 90	D4 = 90
	Avg = 87.25	Avg = 90	Avg = 87	Avg = 85.66
K-Nearest Neighbour	D1 = 61	D1 = 59	D1 = 60	D1 = 61
	D2 = 59	D2 = 59	D2 = 59	D2 = 57
	D3 = 61	D3 = 60	D3 = 61	D3 = 59
	D4 = 69	D4 = 68	D4 = 67	D4 = 67
	Avg = 61.50	Avg = 61	Avg = 60	Avg = 60
Decision Tree	D1 = 77	D1 = 77	D1 = 77	D1 = 77
	D2 = 94	D2 = 95	D2 = 94	D2 = 94
	D3 = 93	D3 = 94	D3 = 94	D3 = 93
	D4 = 85	D4 = 87	D4 = 86	D4 = 86
	Avg = 88	Avg = 89.33	Avg = 89	Avg = 89
Random Forest	D1 = 80	D1 = 80	D1 = 80	D1 = 80
	D2 = 95	D2 = 96	D2 = 95	D2 = 95
	D3 = 93	D3 = 93	D3 = 93	D3 = 93
	D4 = 100	D4 = 100	D4 = 99	D4 = 99
	Avg = 92	Avg = 92.25	Avg = 91.75	Avg = 91.75
Gradient Boosting	D1 = 77	D1 = 70	D1 = 68	D1 = 65
	D2 = 86	D2 = 88	D2 = 87	D2 = 86
	D3 = 84	D3 = 86	D3 = 84	D3 = 84
	D4 = 80	D4 = 83	D4 = 80	D4 = 80
	Avg = 83.33	Avg = 82	Avg = 80.66	Avg = 80

4.3 Answer to RQ. 3

To answer RQ.3 “Which one is efficient opinion mining classifier based on the comparison of different machine learning classifiers?”, we compare the performance evaluation results obtained from different classifiers, when applied on multiple (4) datasets of multiple domains (Table 10).

Classifier with Best Performance

In our work, on all three datasets, Random Forest outperformed other algorithms. The Random Forest is based on an ensemble method which combines several machine learning techniques into one model to reduce variance and bias. The ensemble is a supervised learning technique, which is a blend of learning algorithms. The ensemble is the procedure of using multiple algorithms to acquire improved predictive performance. The ensemble technique uses multiple machine learning classifiers, sometimes uses a combination of different type of classifiers, expecting that the performance is much higher than the individual classifiers that are used.

The Random Forest implements bagging, also called bootstrap aggregating technique to enhance the stability and accuracy of machine learning algorithms [38] Furthermore, it is reported by [28] that Random performs better than the other classifiers on imbalanced datasets, as a small portion of the training set is used by each tree to grow. Random Forest Classifier is based on ensemble method which makes Random Forest more effective than other ML classifiers on Imbalanced datasets. In our study the datasets are imbalanced, which are suitable for Random Forest, and consequently RF performs better when compared to other methods/classifiers used in this study.

The distribution of classes in our dataset is imbalanced, as some classes exist more frequently than the others. For example, the “support” column of Figure 2 shows that there are 243 instances of the *pos_neg* class, which is quite high as compared to that of other classes. On the contrary, *neg_neg* classes have the least number of instances, i.e. 10. As reported by [28] Random Forest performs better for the imbalanced dataset, and similar is the case in our experimental findings, i.e. Random Forest performs better than other classifiers on the imbalanced dataset.

The experimental results presented in Table 8 depict that after Random Forest, Decision tree gives the best results w.r.t the other classifiers used in this work. The decision tree is a machine learning classifier which aims to solve the classification and regressions problems. It at-

93 %	precision	recall	f1-score	support
neg_neg	1.00	0.73	0.84	11
neg_neu	1.00	1.00	1.00	13
neg_pos	1.00	0.86	0.92	69
neu_neg	1.00	1.00	1.00	10
neu_neu	0.88	0.85	0.86	52
neu_pos	0.90	0.93	0.91	28
pos_neg	0.92	0.98	0.95	243
pos_neu	0.96	0.89	0.92	87
pos_pos	0.89	0.97	0.93	87
avg / total	0.93	0.93	0.93	600

Figure 2: Performance evaluation results of Random Forest

tempts to find a solution to the problem by following the tree representation [29].

It is reported by [15] that the decision tree performs better on the imbalanced datasets. Furthermore, work performed by [31] on imbalanced datasets, reported that the predictive accuracy of the Decision Tree Classifier gets improved, when applied on imbalance datasets with multiple classes. In another study, [32] investigated the performance of machine learning classifiers on imbalanced datasets, and it was observed that due to efficient feature selection strategy using recursive and top-down greedy search technique, Decision Tree classifier performs better on imbalanced and multi-class datasets.

As the datasets used in this work are imbalanced with multi-classes, and resultantly the Decision Tree performed better than the other classifiers except for Random Forest, reported earlier. Therefore, our findings are in-line with that of the aforementioned earlier works, i.e. Decision Tree classifier gives better results multi-class imbalanced datasets.

Classifier with Worst Performance

KNN:

Results presented in Table 8 show that KNN gives the worst performance on all of the three datasets. The major reason for such a degraded performance of KNN is due to the issue of “skewed-class”, i.e. it gives poor prediction results on the datasets with the imbalanced distribution of classes [44]. In this work, there is the imbalanced distribution of classes in the three datasets, therefore, KNN suffered from the issue of “skewed-class” and exhibited poor performance (Table 10).

NB:

The NB classifier yielded poor results (Table 10) due to the following reasons: (1) The NB classifier needs a large dataset for producing good results and gives poor performance on a small dataset [33]. In the proposed system, there are three datasets comprising of few thousands of reviews, *i.e.* 6000, 3000, 3000 reviews for the three datasets respectively. Due to this small size, NB produced poor results, (2) The NB classifier gives the worst performance on multi-class datasets, as it is not an efficient predictor on datasets with multi-classes [33]. In the proposed study, there are multiple (9) classes, due to which NB yielded poor performance during classification, and (3) There exist different factors, such as the size of the dataset, skewed-class (imbalanced classes), number of classes, the structure of the dataset, Number of instances that are available in the Dataset and the ratio of splitting of the dataset into Training and Testing [34]. In this study, we observed that the performance of the machine learning classifiers is mostly dependent on the *size of the dataset*, a *number of classes* and *imbalanced (skewed-class)* nature of the dataset.

Experiment on Balanced Dataset

Although the performance of M.L classifiers is degraded by increasing number of classes, however, by balancing the dataset, more robust results are achieved. The results are depicted in Table 10. Firstly, we combined three datasets (D1, D2, D3) to constitute a larger dataset of 12000+ reviews. To balance the resultant dataset, we applied a SMOTE in Anaconda Framework, which is an over-sampling technique [31]. We choose to use over-sampling technique due to its performance efficiency reported by [35]. After balancing the dataset by using the above-mentioned technique, we get improved results, as reported in the Table 12.

The Table 12 shows the results of combined datasets (D1+D2+D3) by applying different machine leaning tech-

Table 12: Performance Evaluation Results on Balanced Dataset

Classifiers	Combined Balanced Dataset (D1+D2+D3)			
	A	Pr	R	F-1
Naïve Bayes	75	76	75	70
Logistic Regression	80	80	78	78
Support Vector Machine	90	91	90	91
<i>K-Nearest Neighbour</i>	65	64	65	64
Decision Tree	96	95	95	95
Random Forest	98	99	98	98
Gradient Boosting	91	91	90	90

niques. The acquired results show the performance of different machine learning algorithms used in this study. Random Forest yielded the best performance outcomes and proves the superiority as compared to other M.L Classifiers.

Suggesting Classifier with Best Classification Results for Comparative Opinion Mining

After the inspection of results presented in Table 10 and Table 11, we recommend Random Forest and Decision Tree for comparative opinion mining on account of their performance being superior with respect to the other machine learning classifiers.

Comparison with baseline methods

In another experiment, the performance of the machine learning classifier Random Forest that yielded the best performance in our work is compared with the efficiency of state-of-the-art studies conducted on comparative opinion mining. Table 12 shows the comparative results. It is evi-

Table 13: Comparison with baseline methods

Study	Methods/Techniques	Results			
		Accuracy (%)	Precision (%)	Recall (%)	F-Score (%)
Ejaz et al. [8]	Lexicon-based approach	89.5	-	-	-
Khan et al. [9]	Naïve Bayes	46.15	41	39	-
Vidhita et al. [10]	Probabilistic Latent Semantic Analysis model	78.9	70	89	90
Our Work	M.L classifiers (R.F)	92 (Avg)	92.25 (Avg)	91.75 (Avg)	91.75 (Avg)

dent that the proposed work (Random Forest) performed better than the comparing methods in term of accuracy, precision, recall, and f-score.

Applying Statistical Tests

In order to validate, whether the Random Forest classifier is the best classifier than the other machine learning classifiers, such as Decision Tree and Support Vector Machine, we have conducted two experiments as shown in Table 14 and Table 15.

We aim to investigate that the performance difference between the Random Forest and the Decision Tree, Random Forest vs Support Vector Machine classifiers, is statistically significant and does not occur randomly. Therefore, we applied a Confidence Interval statistical method [45], which involves two hypothesis *i.e.* null hypothesis and alternate hypothesis given as follows:

H₀: The error rate of the two classifiers is the same. **H_A:** The error rate of both classifiers is significantly different.

Table 14 and Table 15 depict the Significance Tests to measure the performance difference between Three Classifiers, namely Random Forest, Decision Tree and SVM.

Table 14: Random Forest vs Decision Tree Comparison

Classifier	N (Number of Reviews)	Error	
		Rate	Difference
Random Forest	15000	0.08	0.332
Decision Tree	15000	0.12	0.467

Table 15: Random Forest vs SVM Comparison

Classifier	N (Number of Reviews in Dataset)	Error	
		Rate	Difference
Random Forest	15000	0.08	0.233
Support Vector Machine	15000	0.11	0.0366

Discussion

In Table 14, the statistical result demonstrates that the error rate of both classifiers is different. Thus, it proves that there is a significant difference between the Random forest and Decision tree methods. Similarly, the statistical results of Table 15 depict that due to the distinct error rate of classifiers, there is a significant difference between the two classifiers. So, a null hypothesis is rejected, and thus the alternate hypothesis is accepted. Hence this proved that the Random forest is statistically significant.

After performing the aforementioned statistical analysis by using the method of “Confidence Interval”, the significance results show the improved performance of the proposed model (Random Forest) for Online Comparative User Reviews.

Applying T-test

To verify statistically that whether the Random Forest classifier is the best classifier when compared to the other machine learning classifiers, such as Decision Tree, we employed t-test to observe whether the difference between the classifiers is statistically significant. For this purpose, t-test is applied on the accuracies of both (best) classifiers, namely “*Random Forest and Decision Tree*”. It is observed that the difference between the classifiers is statistically significant. The results are presented in the Table 16 and demonstrated via a graphical representation (Figure 3).

Table 16: Results of the t-test

t-value	p-value
−5.8286	1.60379E−0.5

Figure 3: Graph-based representation

Python-based (sci.py) Anaconda Framework is used to implement the desired t-test.

5 Conclusion and Future work

This work deals with the performance-based comparison of different machine learning classifiers with respect to comparative opinion mining. It performs the following tasks: (1) Acquisition of the three datasets with respect to comparative opinion mining, (2) cleaning of text by applying different pre-processing steps, (3) performing comparative opinion mining using different machine learning classifiers, (4) comparing the results of classifiers used in this study, and (5) recommendation of the classifier with efficient results.

The proposed method aimed at classifying the comparative reviews into different polarity classes: “pos_pos, pos_neg, pos_neu, neg_neg, neg_pos, neg_neu, neu_neu, neu_pos and neu_neg”. The machine learning classifiers applied in this study include Naïve Bayes, Logistic Regression, Support Vector Machine (SVM), KNN (K-Nearest Neighbor), Decision Tree, Random Forest, and Gradient Boosting. The input review is classified into nine different sentiment classes. Different performance evaluation metrics are applied to analyze the effectiveness of different machine learning classifiers. The “Random Forest” yielded best results (P: 96, R: 95, F-score: 95, and A: 95), whereas the KNN classifier yielded poor performance with respect to comparing algorithms.

Limitations: (1) We used three datasets for performing experiments, however, the size and number of datasets are insufficient for obtaining robust results, (2) the datasets used in this study are imbalanced which results in performance degradation of the machine learning classifiers. [3], (3) It is observed that due to the increased number of classes (nine) used as polarity labels in the acquired datasets, the performance of the machine learning classifiers for classifying comparative reviews is degraded [36]. (4) The domain of the three datasets used in the study is the same, i.e. “technological products” such as *iOS vs Android*, *Microsoft vs Google*, and *Facebook vs Twitter*. Therefore, due to the same domain of the datasets used, the robustness of the classifier is questionable, needing further experimentation, and (5) The random splitting method is used for datasets segmentation into training and testing.

Future Directions: (1) Multiple datasets with increased in size can be experimented with to obtain more reliable re-

sults, (2) Performance of the machine learning classifier can be further improved by using balanced datasets, (3) Other experiments can be carried out by changing the number of classes i.e. reducing the classes from nine to other numbers, can bring more improved results, (4) The performance of the machine learning classifiers can be experimented on and evaluated using datasets in multiple domains with respect to comparative opinion mining, and (5) other techniques of dataset splitting such as cross-validation, hold-out cross-validation, stratified sampling, and some others, can be applied to evaluate the performance of machine learning classifiers.

Conflict of Interests: Authors declare that they have no conflict of interests.

References

- [1] Liu, B., 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1), 1-167.
- [2] Varathan, K. D., Giachanou, A., & Crestani, F., 2017. Comparative opinion mining: a review. *Journal of the Association for Information Science and Technology*, 68(4), 811-829.
- [3] Wang, S., & Yao, X., 2012. Multiclass imbalance problems: Analysis and potential solutions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(4), 1119-1130.
- [4] Jindal, N., & Liu, B., 2006. Mining comparative sentences and relations. In *AAAI* (Vol. 22, No. 13311336, p. 9).
- [5] Xu, K., Liao, S. S., Li, J., & Song, Y., 2011. Mining comparative opinions from customer reviews for Competitive Intelligence. *Decision support systems*, 50(4), 743-754.
- [6] Bach, N. X., Van, P. D., Tai, N. D., & Phuong, T. M., 2015. Mining Vietnamese comparative sentences for sentiment analysis. In *Knowledge and Systems Engineering (KSE), Seventh International Conference on* (pp. 162-167). IEEE.
- [7] Duriqi, R., Raca, V., & Cico, B., 2016. Comparative analysis of classification algorithms on three different datasets using WEKA. In *Embedded Computing (MECO), 2016 5th Mediterranean Conference on* (pp. 335-338). IEEE.
- [8] Ejaz, A., Turabee, Z., Rahim, M., & Khoja, S., 2017. Opinion mining approaches on Amazon product reviews: A comparative study. In *Information and Communication Technologies (ICICT), International Conference on* (pp. 173-179). IEEE.
- [9] Khan, A. U. R., Khan, M., & Khan, M. B., 2016. Naïve Multi-label classification of YouTube comments using comparative opinion mining. *Procedia Computer Science*, 82, 57-64.
- [10] Vidhita Kothari, Arindam Chaudhuri, & Saifee Vohra., 2016. Contrastive Opinion Summarization based on Blog Threads. *International Journal Of Advance Research And Innovative Ideas In Education*, 2(3), pp 3761-3769.
- [11] Rana, S., & Singh, A. (2016) Comparative analysis of sentiment orientation using SVM and Naive Bayes techniques. In *Next Generation Computing Technologies (NGCT), 2nd International Conference on IEEE*. pp. 106-111.

- [12] Joshi, R., & Tekchandani, R., 2016. Comparative analysis of Twitter data using supervised classifiers. In *Inventive Computation Technologies (ICICT)*, International Conference on IEEE (Vol. 3, pp. 1-6).
- [13] Devika, M. D., Sunitha, C., & Ganesh, A., 2016. Sentiment analysis: A comparative study on different approaches. *Procedia Computer Science*, 87, 44-49.
- [14] Nayak A., Nataranjan S., 2016. Comparative study of Naïve Bayes, Support Vector Machine and Random Forest Classifiers in Sentiment Analysis of Twitter feeds. *International Journal of Advanced Studies in Computer Science and Engineering (IJASCSE)* Issue 1, Volume 5.
- [15] Medagoda, N., Shanmuganathan, S., & Whalley, J., 2013. A comparative analysis of opinion mining and sentiment classification in non-English languages. In *Advances in ICT for Emerging Regions (ICTer)*, International Conference on (pp. 144-148). IEEE.
- [16] Ganapathibhotla, M., & Liu, B., 2008. Mining opinions in comparative sentences. In *Proceedings of the 22nd International Conference on Computational Linguistics Association for Computational Linguistics..* Volume 1. pp. 241-248.
- [17] Ghag, K., & Shah, K., 2013. Comparative analysis of the techniques for Sentiment Analysis. In *Advances in Technology and Engineering (ICATE)*, International Conference on (pp. 1-7). IEEE.
- [18] Bhavitha, B. K., Rodrigues, A. P., & Chiplunkar, N. N., 2017. Comparative study of machine learning techniques in sentimental analysis. In *Inventive Communication and Computational Technologies (ICICCT)*, International Conference on IEEE. pp. 216-221.
- [19] Lerman, K., & McDonald, R., 2009. Contrastive summarization: an experiment with consumer reviews. In *Proceedings of human language technologies: The annual conference of the North American chapter of the association for computational linguistics, companion volume: Short papers* (pp. 113-116). Association for Computational Linguistics.
- [20] Jin, J., Ji, P., & Gu, R., 2016. Identifying comparative customer requirements from product online reviews for competitor analysis. *Engineering Applications of Artificial Intelligence*, 49, 61-73.
- [21] McHugh, M. L., 2012. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3), 276-282.
- [22] Tarang Shah., About Train, Validation and Test sets in Machine Learning, available at: <https://towardsdatascience.com/train-validation-and-test-sets-72cb40c9e7>. Last accessed: December 15, 2018.
- [23] Reitermanova, Z., 2010. Data splitting. In *WDS* (Vol. 10, pp. 31-36).
- [24] Nabil, M., Aly, M., & Atiya, A., 2015. Astd: Arabic sentiment tweets dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 2515-2519).
- [25] Asghar, M. Z., Khan, A., Khan, F., & Kundi, F. M., 2018. RIFT: A Rule Induction Framework for Twitter Sentiment Analysis. *Arabian Journal for Science and Engineering*, 43(2), 857-877.
- [26] Effrosynidis, D., Peikos, G., Symeonidis, S., & Arampatzis, A., 2018. DUTH at SemEval-2018 Task 2: Emoji Prediction in Tweets. In *Proceedings of the 12th International Workshop on Semantic Evaluation* (pp. 466-469).
- [27] B., Guo, Y., & Chen, X., 2017. Mining User Reviews for Mobile App Comparisons. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3), 75.
- [28] Chen, C., Liaw, A., & Breiman, L., 2004. Using random forest to learn imbalanced data. University of California, Berkeley, 110, 1-12.)
- [29] Rahul Saxena, "How Decision Tree Algorithm Works", available at: <http://dataaspirant.com/2017/01/30/how-decision-tree-algorithm-works>. Last accessed (November 15, 2018).
- [30] Medagoda, N., & S., & Whalley, J., 2013. Analysis of classification in multiple languages. In *Advances in ICT for Emerging Regions (ICTer)*, International Conference on (pp. 170-182). IEEE.
- [31] Chawla, N. V., 2009. Data mining for imbalanced datasets: An overview. In *Data mining and knowledge discovery handbook* (pp. 875-886). Springer, Boston, MA, in his work
- [32] He, H., & Garcia, E. A., 2008. Learning from imbalanced data. *IEEE Transactions on Knowledge & Data Engineering*, (9), 1263-1284,
- [33] Rish, I., 2001. An empirical study of the naive Bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence* (Vol. 3, No. 22, pp. 41-46). New York: IBM
- [34] Van der Walt, C. M., & Barnard, E., 2006. Data characteristics that determine classifier performance.
- [35] Kaur, P., & Gosain, A. (2018). Comparing the Behavior of Oversampling and Under sampling Approach of Class Imbalance Learning by Combining Class Imbalance Problem with Noise. In *ICT Based Innovations* (pp. 23-30). Springer, Singapore)
- [36] Silva-Palacios, D., Ferri, C., & Ramírez-Quintana, M. J., 2017. Improving Performance of Multiclass Classification by Inducing Class Hierarchies. *Procedia Computer Science*, 108, 1692-1701.
- [37] Holte, R. C., 1993. Very simple classification rules perform well on most commonly used datasets. *Machine learning*, 11(1), 63-90.
- [38] Ismail, H., Harous, S., & Belkhouche, B., 2016. A Comparative Analysis of Machine Learning Classifiers for Twitter Sentiment Analysis. *Research in Computing Science*, 110, 71-83.
- [39] Joachims, T., 1998. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning* (pp. 137-142). Springer, Berlin, Heidelberg.
- [40] Leonard Richardson., "Beautiful Soup", available at: <https://www.crummy.com/software/BeautifulSoup>. Last accessed (October 22, 2018).
- [41] Li, Y., Jia, B., Guo, Y., & Chen, X., 2017. Mining User Reviews for Mobile App Comparisons. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3), 75.
- [42] N. V. Chawla, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Oversampling Technique. *Journal of Artificial Intelligence Research*, 16:321-357, 2002.
- [43] Wang, W., Xin, G., Wang, B., Huang, J., & Liu, Y., 2017. Sentiment information Extraction of comparative sentences based on CRF model. *Computer Science & Information Systems*, 14(3).
- [44] Zhang, M. L., & Zhou, Z. H., 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7), 2038-2048.
- [45] Box, G. E., Hunter, W. G., & Hunter, J. S. (1978). *Statistics for experimenters*