Sentiment Analysis of Product Reviews to Identify Deceptive Rating Information in Social Media: A SentiDeceptive Approach

M. Irfan Marwat[1], Javed Ali Khan[1], Dr. Mohammad Dahman Alshehri[2*], Muhammad Asghar Ali[1], Hizbullah[1], Haider Ali[1] and Muhammad Assam[3]

[1] Department of Software Engineering, University of Science and Technology Bannu, at Bannu, KPK 28100 Pakistan
[e-mail: engr_javed501@yahoo.com]
[2] Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia
[e-mail: alshehri@tu.edu.sa]
[3]College of computer Science, Zheijiang University, Hanghzou, China
[*]Corresponding author: Dr. Mohammad Dahman Alshehri

Received September 2, 2021; revised January 2, 2022; accepted March 9, 2022; published March 31, 2022

Abstract

[**Introduction**] Nowadays, many companies are shifting their businesses online due to the growing trend among customers to buy and shop online, as people prefer online purchasing products. [Problem] Users share a vast amount of information about products, making it difficult and challenging for the end-users to make certain decisions. [**Motivation**] Therefore, we need a mechanism to automatically analyze end-user opinions, thoughts, or feelings in the social media platform about the products that might be useful for the customers to make or change their decisions about buying or purchasing specific products. [**Proposed Solution**] For this purpose, we proposed an automated SentiDecpective approach, which classifies end-user reviews into negative, positive, and neutral sentiments and identifies deceptive crowd-users rating information in the social media platform to help the user in decision-making. [**Methodology**] For this purpose, we first collected 11781 end-users comments from the Amazon store and Flipkart web application covering distant products, such as watches, mobile, shoes, clothes, and perfumes. Next, we develop a coding guideline used as a base for the comments annotation process. We then applied the content analysis approach and existing VADER library to annotate the end-user comments in the data set with the identified codes, which results in a labelled data set used as an input to the machine learning classifiers. Finally, we applied the sentiment analysis approach to identify the end-users opinions and overcome the deceptive rating information in the social media platforms by first preprocessing the input data to remove the irrelevant (stop words, special characters, etc.) data from the dataset, employing two standard resampling approaches to balance the data set, i-e, oversampling, and under-sampling, extract different features (TF-IDF and BOW) from the textual data in the data set and then train & test the machine learning algorithms by applying a standard cross-validation approach (KFold and Shuffle Split). [**Results/Outcomes**] Furthermore, to support

http://doi.org/10.3837/tiis.2022.03.005 **ISSN : 1976-7277**

our research study, we developed an automated tool that automatically analyzes each customer feedback and displays the collective sentiments of customers about a specific product with the help of a graph, which helps customers to make certain decisions. In a nutshell, our proposed sentiments approach produces good results when identifying the customer sentiments from the online user feedbacks, i-e, obtained an average 94.01% precision, 93.69% recall, and 93.81% F-measure value for classifying positive sentiments.

Keywords: Sentiment Analysis, Opinion Mining, Customer Reviews, Natural Language Processing, Imbalance, Deceptive reviews, Flipkart, Amazon

1. Introduction

For a decade, the internet has gained access in every domain or area and has become an integral part of all walks of life. One such example is the emergence of Cloud computing and the internet of things (IoT) [51, 52, 53, 54]. Recently, customers started registering their feelings and thoughts over the internet in using rating and text (reviews) against a specific product. All users' reviews are stored as a vast amount of essential data for each hour, day, and week. For example, in an empirical research study, it is identified that mobile software applications receive roughly 23 reviews per day while more popular apps, i-e, Facebook, received almost 4275 user feedbacks per day [1]. Similarly, end-users register approximately 30,000 tweets for popular software applications, such as Snapchat and Facebook [2]. Furthermore, user reviews play a pivotal role in deciding to purchase or buy a specific product. However, the truthfulness and authenticity of these users' reviews are still not guaranteed, and these reviews, new websites, and content platforms are susceptible to deceptive information [3]. For example, the user reviews where text represents the negative review. Still, the rating is 5-stars in the dataset, i.e., *"very bad use chattering work dress"*, *"the worst product ever it is not separate it is attached with it you cannot wear the only shirt it is very bad"*. Analyzing users' reviews will enhance both the customers and the companies. Sentiment analysis has gotten its identification and is used to classify the reviews [4].

The end-users opinions and comments expressed in the online user forums, e-commerce websites, social media applications, and app stores primarily attract and influence potential customers in making certain decisions. A study reported that 87% of customers or end-users change their purchase or buying decisions after reading positive feedback. In comparison, 80% of customers change their purchasing decision after confronting or reading negative reviews [5]. In contrast, the customer accuracy in identifying deceptive opinions in online user reviews is 61.9 % [6]. Although these user opinions in the social media platforms are not considered as malware, such deceptive end-user opinions can result in a privacy and security risk by forcing potential users to purchase a specific company product or service by giving artificial positive feedback or discouraging potential online customers from buying product or services when registered negative feedback against it.

It inspired us to propose an automated sentiment analysis-based approach that identifies user opinions based on end-user comments in the social media platforms, i-e, Amazon and Flipkart, to identify deceptive rating information and help potential customers in decision-making. Sentiment analysis is a process of analyzing data based on the person's feelings, reviews, thoughts, and emotions about some specific product or service [7]. Sentiment analysis is also called opinion mining because it extracts important features from end-user reviews. Sentiment analysis was done using machine learning algorithms, models, techniques, and natural language processing, which extract essential features from large datasets. Sentiment Analysis can be performed at document, sentence, and phrase levels [7]. At the document level, a summary of the entire document is taken first and then analyzed whether the sentiment is positive, neutral, or negative. Each sentence is classified in a particular class (positive, neutral, or negative) at the sentence level to provide the sentiment. In contrast, analysis of phrases in a sentence is taken into account to check polarity at the phrase level [38]. The author's analyzed user feedback at the review level for this approach, also referred to as document level, where each end-user review identified its sentiment.

Amazon[1] and Flipkart[2] are online shopping platforms where different companies offer their products for sale. The customers posted reviews about the purchased products according to their feelings, satisfaction, opinions, and thoughts. Many data (end-user reviews) are unstructured and written in natural language [55, 56]. While manually processing and analyzing such a large number of end-users reviews is time-consuming, costly, and requires much effort to identify helpful information for potential customers. Therefore, a sentimental analysis and opinion mining-based approach is utilized to process the end-reviews for a particular product and predict the sentiment of each review, either positive, neutral, or negative, by employing different Machine Learning. Finally, the best results can be added to the existing online shopping platforms to help potential customers make reliable decisions and give a clearer picture of product performance based on end-user reviews.

Our contribution in this paper includes the duration of a new research data set from Amazon and Flipkart containing end-users reviews about different products; we experiment with different machine learning classifiers for the automated classification of user comments and compared their performance in the social media data set; finally, develop a research-based tool that implements the best performing machine learning classifier and display an information chart containing supporting, attacking, and neutral reviews about a specific product that helps to overcome the deceptive rating information.

To accommodate such end-user generated feedback into the existing online shopping platforms, we conducted an experimental study to answer the following research questions:

RQ-1: Is sentiment analysis helps to identify deceptive rating information in social media platforms?

RQ2: What machine learning classifiers can be used to identify the sentiments of user comments or reviews and how well they performed when analyzing social media data?

The main structure of the research paper is as follows:

Section 2 introduces the related work on the sentiment analysis; section 3 highlights the proposed research methodology, section 4 discusses the curated research data set, section 5 elaborates the annotation and coding process, section 6 describe the sentiment analysis experiment and natural language processing steps, section 7 discuss the proposed

[1] https://www.amazon.com accessed on 1-10-2021
[2] https://www.flipkart.com accessed on 1-12-2021

SentiDeceptive tool and its working, section 8 discusses the overall process of the SentiDeceptive tool and section 7 concludes the paper, discuss the limitation and highlight the future work.

2. Literature Survey

Sentiment analysis of Product review describes the customers' sentiment about a product based on their reviews. In this section, we reported on the recent work on the sentiment analysis in different platforms.

2.1 Machine Learning Based Sentiment Analysis

Singla et al. [8] present an automated approach by experimenting with over 4,000,00 reviews and used sentiment analysis to classify the reviews into positive and negative classes. The models are performed with 10-Fold Cross-Validation, and classification or machine learning models used for their research include Naïve Bayes, Support Vector Machine (SVM), and Decision Tree. SVM gives the best accuracy than Naïve Bayes and Decision Tree, 81.77%. Dey et al. [9] use the Amazon dataset for sentiment analysis in their research. The dataset contains almost 1,47,000 reviews of the books. The classification is binary means positive and negative class. The classification is based on review rating, and the reviews with 5 and 4 ratings are considered positive. Reviews with rating 3 discards from the dataset and reviews with ratings 2 and 1 are considered negative. Their research preprocessing includes tokenization, removing stop words, and filling the missing value with global or universal constant. Moreover, the feature selection includes TF-IDF, frequent noun identifier, and relevant noun removal. SVM and Naïve Bayes machine learning classifiers or models used to classify the review positively or negatively. SVM provides high accuracy compared to Naïve Bayes, which is 84%. Haque et al. [10] proposed a machine learning model that polarises reviews and learns from them. They used the machine learning classification models on a large-scale amazon dataset to polarize it and get acceptable or justifiable accuracy. The dataset was categorized into three Electronics reviews, cell phone, accessories reviews, and musical instruments. The total reviews are approximately 48500, where 21600 reviews are from mobile phones, 24352 are from electronics & 2548 are from musical instruments data. They performed sentiment analysis on a rating level, where 5 and 4 rating reviews were considered positive, 3 rating reviews were deemed neutral, and 2 and 1 rating reviews were considered negative. Preprocessing includes tokenization, removal of stop words, and POS tagging, and feature selection includes bag-of-word and TF-IDF methods. The research use six machine learning classification models, which are Naïve Bayesian, Support vector Machine Classifier (SVC), Stochastic Gradient Descent (SGD), Linear Regression (LR), Random Forest, and Decision Tree. K-fold cross-validation is used for training and testing the machine learning model. In k-fold cross-validation, they utilize 10-fold cross-validation. The highest accuracy achieved by SVM is 94.02%.

Rain [11] extends the latest work in sentiment analysis and natural language processing to data revive or retrieve from Amazon. The dataset used in this research contains customer product reviews. The dataset includes 50,000 user reviews from 15 different products. The number of stars a client gives to an item is utilized as preparing information to perform supervised machine learning. They used two machine learning classification models: Naive Bayes and decision list, which were used to classify a given review into positive or negative.

The Naïve Bayes gives the highest accuracy than the decision list, 0.8449%. Furthermore, Sandeep et al. [12] use the Amazon review dataset for sentiment analysis. They use characteristics from the document matrix using the bi-gram modelling technique. The review sentiment is categorized into positive and negative reviews. They remove unique characters and numeric values in preprocessing and use the Snowfall Stemming approach. After that term frequency, each word is recorded with the Word Sack, which displays documents and counts the number of words that appears in the text (document term matrix). The next step is to split a dataset into test and train datasets using cross-validation, 90% for training and 10% for testing. They use three machine learning algorithms: Linear SVC, Voting, and Naïve Bayes. Linear SVC gives the highest accuracy, which is 91.00%. In the end, they draw the ROC curve for each algorithm.

Lakshmi et al. [13] proposed sentiment analysis on the Flipkart dataset. The classification is at a document level. They classify the user review into positive and negative sentiments. They employ three (3) machine learning algorithms, which are Naïve Bayes, Decision tree, and one deep learning model, i-e, neural network. They achieved 0.90% accuracy with the neural network algorithm. Venkataet et al. [14] experimented with the end-users reviews extracted from the Flipkart online shopping website. They identified the customer opinions by combining four parameters: star ratings of the product, the polarity of the review, age of review, and helpfulness score. Kaur and Singla [15] present an explanatory study of the efficacy of classifying product reviews by semantic meaning. They propose entirely different approaches, including spelling correction in review text and classifying reviews using a hybrid algorithm combining Decision Trees and Naive Bayes algorithms. Yasen and Tedmori [16] use the IMDB dataset and classify the user reviews into positive and negative. The methodology includes word tokenization, word filtering, Stemming, and Attribute Selection. They use eight different machine learning algorithms, such as Naïve Bayes, Decision Tree, SVM, Bayes Network, K-nearest Neighbors, Ripper Rule Learning, Random Forest, and Stochastic Gradient Descent. The Random Forest classifier achieved the best accuracy that is 96.01%. Furthermore, Random Forest also got the precision of 0.93%, f-measure 0.96%, and AUC 0.96%. Ahmed et al. [17] propose that the count of scored opinion words is classified into seven possible categories, i.e., strong-positive, positive, weak-positive, neutral, weak-negative, negative, strong-negative. They performed the sentiment analysis by intertwining the score counts. For this purpose, they use different machine learning algorithms, i-e, SVM, Multilayer Perception (MLP), and Naïve Bayes. The MLP and Naïve Bayes classifiers outperform the SVM classifier. Boiy et al. [18] proposed an approach that identifies users' sentiments about specific topics in the social media platform. In contrast, the sentiments can be positive, negative, or neutral. They experimented with Symbolic Techniques and Machine Learning Techniques. They achieved 84.0% and 76.72% accuracy with Symbolic Techniques, while maximum accuracy of 87.40% is achieved with the machine learning classifiers.

2.2 Deep Learning Based Sentiment Analysis

AlQahtani [57] proposed a research study on the sentiment analysis of the Amazon end-reviews at a different level of granularity, i-e, binary and Multiclass classification. Their curated dataset comprises more than 400,000 end-reviews covering different mobile phone categories. They use different techniques to transform the textual data into vectors representation, i-e., Bag-Of-Words, Tf-IDF, and Glove. Finally, they trained and validated various Machine and deep learning algorithms such as Logistic Regression, Random Forest, Naïve Bayes, Bidirectional Long-Short Term Memory, and Bert to identify end-user

sentiments in the amazon store. The BERT classifier achieved the highest accuracy of 98% and 94.7%, respectively, in binary and multiclass classification. Shah [58] proposed a binary sentiment-based approach that classifies end-user reviews positively and negatively. The proposed sentiment analysis-based approach is based on ratings. The Reviews above 3-star ratings have assigned Value 1 representing positive Sentiment and Value 0 expressing negative Sentiment. The preprocessing includes removing punctuation and stop words, case conversion and lemmatization, TF-IDF, Bag of Words and FastText Word embedding. Finally, different machine and deep learning algorithms, including Naïve Byes, Support Vector Machine (SVM), Convolutional Neural network (CNN), are used and compared the performance of word2vec-CNN Model with FastTextCNN Model on amazon unlocked mobile phone dataset. The FastText-CNN Model achieved the highest accuracy that is 0.9462%. Nandwani and Verma [59] proposed a literature study that elaborates on the process of sentiment analysis and emotion detection from social media text. The recommended preprocessing steps performed in the literature for sentiment analysis are tokenization, stop word removal, POS tagging, Stemming and lemmatization. The recommended feature extraction includes Bag of Words (BOW), N-gram, TFIDF, and word embedding. Finally, the popular techniques used for sentiment analysis and emotion detection are the lexicon-based approach, machine Learning-based approach, deep Learning-based approach, transfer Learning approach, and hybrid approach.

2.3 Product or Services Recommendations Based on Sentiment Analysis

Hu et al. [60] proposed a heuristic-based recommendation approach for end-user interest profiling. The proposed approach comprises end-user feature extraction, assesses reviewer credibility to restrict fake reviews, mining end-user interests, identifying sentiment score using fastText, and finally recommended purchase feature using sentiment score. The proposed approach's mean average precision (MAP@1) is 93%, and MAP@3 is 49%, respectively. Zhao et al. [61] proposed a sentiment-based location recommender system by utilizing sentimental attributes of desired locations by employing a point of interest mining method rather than only mining end-user check-in information mining. Furthermore, Zhao et al. [60] proposed an Emoji recommendation system that recommends appropriate Emoji between the thousand other emojis by utilizing contextual and end-user personal information. The authors extensively experimented with different data sets and concluded that the proposed approach performed better than the existing state-of-the-art Emoji recommendation systems. Additionally, Zhao et al. [61] proposed a product and services recommendation system based on interpersonal influences. For this purpose, they utilized end-user sentimental deviations and crowd-user feedback reliability. The proposed approach could help understand the end-user behaviors and thus improve the product and service recommendations.

The proposed work complements the research mentioned above approaches in sentiment analysis but aims at different perspectives. For example, we are interested in overcoming the deceptive user rating by designing a research approach that identifies end-user sentiments based on original crowd-user comments rather than their associated rating. Secondly, to test and validate our proposed research approach, we aim for generalization by curating a research data set from multiple social media platforms, i-e, Amazon and Flipkart. Finally, to support our research study, we develop a tool that validates our research findings.

3. Proposed Research Methodology

The proposed research methodology is depicted in **Fig. 1**, which comprises four main steps. In the first step, namely "data collection", we curated a research dataset containing end-users reviews against different products, i-e, shoes, watches, clothes, etc., in the social media and e-commerce platforms, such as Amazon and Flipkart. The details of curating a research data set are elaborated in section 4 of the research paper. In the second step, to make the research data pursuable for the machine learning algorithms and automatically identify end-user sentiments in the social media platforms, we develop a coding guideline iteratively that would be used as a base for the annotation process (explained in section 5). Next, using the developed coding guidelines, we applied the content analysis approach to annotate the crowd-users comments in the research data set with the captured codes, i-e, positive, negative, and neutral. Remove the conflicts between the coders and develop a final labelled data set to input the machine learning algorithms. Also, as shown in **Fig. 1**, we utilized the built-in NLP library called Valence Aware Dictionary and sEntiment Reasoner (VADER) to automatically identify the expected label (positive, negative, or neutral) of the end-user comments in the social media platforms. The aim is to identify the performance of the VADER library in automatically identifying the labels of end-user comments and automate the hectic and time-consuming activating of the sentiment annotation process. Thus, we employ two approaches in the proposed SentiDecptive approach to annotate the end-user comments in the dataset, i-e, the manual annotation process and the NLP VADER library. Section 5 of the research paper elaborates the end-user comments annotation steps. Finally, in the fourth step of the proposed SentiDecptive approach, we apply the sentiment analysis approach to identify the end-users opinions and overcome the deceptive rating information in the social media platforms by first preprocessing the input data to remove the irrelevant data, i-e, stop words, special characters, etc., from the dataset. Next, to identify reliable results with the machine learning algorithms, we employ two standard resampling approaches to balance the data set, i-e, oversampling and under-sampling. Next, we extract different features (TF-IDF and BOW) from the textual data in the data set and then train & test the machine learning algorithms by applying a standard cross-validation approach (KFold and Shuffle Split). Finally, we calculate each machine learning classifier's accuracy, precision, recall, and F1 score and report the results to identify the best machine-learning algorithm and overcome the deceptive end-user rating information in the social media platforms. We explain the proposed research methodology in detail in the following sections and sub-sections.

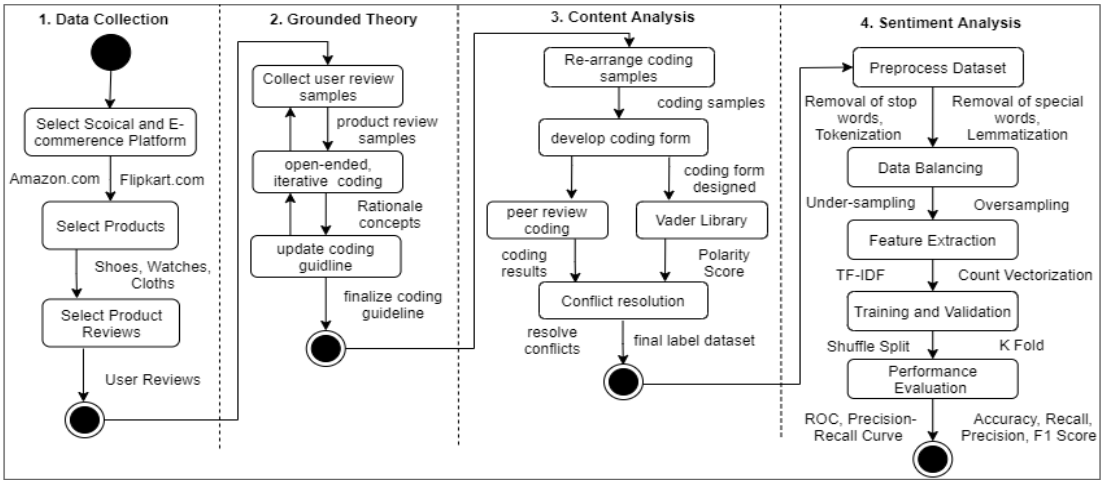

Fig. 1. Proposed System

4. Dataset Collection and Preparation

We curated two datasets from Amazon and Flipkart using a customized web scraping tool to run our proposed SentiDecpective approach. Also, we collected some end-user reviews from these social media platforms manually. Web scraping is a method to extract the structured web data (information) in an automated trend. The extracted information (end-user reviews) is exported into different formats, such as spreadsheets or CSV. Moreover, after collecting the end-user comments from these two online shopping platforms, we merge them into a single dataset. The details about the end-user reviews collected from these two shopping platforms are described in **Table 1**. Furthermore**,** Amazon and Flipkart are the prime e-commerce websites; it is possible to see and collect innumerable user reviews and opinions about different products. Below we discuss these two platforms:

Amazon.com: Amazon is an American multinational organization that focuses on e-commerce business. Amazon has become the largest e-commerce retailer and one of the most potent shopping brands globally. Recently, Amazon has more than 300 million active users, who submit more than thousands of reviews every day.

Flipkart.com: Flipkart is an Indian e-commerce company that enable millions of consumer, seller and small businesses to be a part of India's e-commerce revolution. Furthermore, Flipkart has more than 200 million registered customers, where approximately thousands of reviews get registered by end-users every month.

Considering the importance of the above-mentioned online shopping platforms, we collected end-user comments representing different products and their corresponding product categories or types, as shown in **Table 1**. In total, we collected 11781 end-users reviews from these two platforms covering distant products, such as watches, mobile, shoes, clothes, and perfumes. Each product included in the data set contains one or more product categories. For example, "shoes" have four categories: Adidas, derby, hush puppies, and bata. Whereas each product category has one or more sub-categories, such as walking shoes, derby for men, essence slip-on for men, and men running shoes, we collected 1117, 1027, 1008, 1190 end-user comments, respectively, as shown in **Table 1**. Altogether, the end-user reviews collected from the Amazon and Flipkart shopping platforms represent the main products, such as shoes, watches, etc. A similar process is repeated for the other products included in the data set.

Table 1. Research data set details

Products	Product Categories	Product Sub-Categories	Total Reviews	Platform
Watches	FastTrack	Tees Analog Watch	1100	Flipkart.com
	Lois Caron	Analog Watch	1174	Amazon.com
Mobile accessories	USB Cable	Micro USB Cable	972	Flipkart.com
Shoes	adidas	Walking shoes	1117	Flipkart.com
	Derby	Derby For Men	1027	Amazon.com
	Hush Puppies	Essence Slip On For Men	1008	Flipkart.com
	BATA	Men Running shoes	1190	Amazon.com
Clothes	Saara	Graphic Print	1011	Amazon.com
	Xee'	Xee' Slim Men Black	1122	Flipkart.com
	Polo	Men Polo Neck White-Black	981	Fliphart.com
Perfume	Gucci	Women Eau de Perfume	1079	Flipkart.com
		Total End-user Reviews	**11781**	

Furthermore, the data set comprises five columns, i-e, Serial, Product, Product Category, Review Text, and Rating. Each of the columns is defined as:

- **Serial:** Unique number of each review
- **Product**: represents the item's name selected for the sentiment analysis, such as Watches, cloth, Shoes, etc.
- **Product Categories:** Define the sub-types or items of the product included in the data set; for example, the product "Clothes" has three sub-categories, i-e, graphic print, xee' slim men black, and men polo neck white-black.
- **Product sub-categories:** Represents the sub-type of the product categories; for example, in our data set, we selected "Shoes" as a product with four categories, such as Adidas, hush puppies, Derby, and Bata and for each category, we selected a sub-category, i-e, walking shoes for Adidas, etc.
- **Review Text:** Represents the end-user opinion about the specific product in Amazon and FlipKart shopping platforms.
- **Rating:** Represents the rating given by the end-user to a specific product in Amazon and FlipKart shopping platforms based on one's experience while using the product.

An example instance from the curated data set is shown in **Table 2**, where an end-user submitted an opinion "waste product" against a product item "Tees analog watch" that is a sub-type of product category "FastTrack", representing "watches" product**.**

Table 2. An end-user review instance in the data set

Serial	1
Product	watches
Product Categories	FastTrack
Product Sub-Category	Tees Analog Watch
Review Text	Waste Product
Rating	1

5. Types of end-user sentiments and dataset Annotation

We performed two activities in this step of the proposed SentiDeceptive research method; firstly, we identified different sentiment elements in the user reviews using grounded theory [19]. Secondly, we annotate each user comment in the data set using the content analysis approach [20] with previously identified sentiment elements. The details are given below:

Table 3. Identified SentiDeceptive labels from Amazon and FilpKart shopping platforms

S. No	User Review Examples	Identified Label	Label Defination
1.	Worst quality material. Do not buy.	Neg	The end-user submits a negative or attacking comment in response to the product under discussion on Amazon or FilpKart platform.
2.	just ok under this price	Nue	The end-user submits a neutral comment in response to the product under discussion on Amazon or FilpKart platform, whose do not affect the ongoing end-user discussion.

3.	A good one in this range of FastTrack	Pos	The end-user submits a positive or supporting comment in response to the product under discussion on Amazon or FilpKart platform.

5.1 Types of User Sentiments

In this step, we thoroughly analyzed the user comments in our data set to identify and capture the distant sentiments elements using the grounded theory approach [19]. Grounded theory is a systematic and qualitative approach that helps design a theory based on the uniform patterns in the research data set. Therefore, during the analysis of the user comments in the data set, we grouped those user reviews that co-relate into different concepts to make a grounded theory for identifying end-user sentiments in the shopping websites. The detailed analysis of the user comments in the data set results in identifying the following concepts that are also elaborated with examples in the coding guideline[3], i-e, positive claim, negative claim, and neutral claim. The definition of each concept captured from the data set is elaborated with examples in **Table 3**.

5.2 Data Set Annotation using manual content analysis approach

After identifying different labels for the SentiDeceptive approach, we must annotate the end-user comments in the curated data set. Therefore, demonstrating and running the proposed approach requires a manually user annotated dataset that will serve as a truth set to answer RQ2. For this purpose, the first three authors manually annotated the dataset using the content analysis technique [20]. To this, they examined and analyzed in detail each user review gathered from Amazon and Flipkart in our dataset. The annotation process takes almost 24 hours to annotate all the user reviews in the dataset. Furthermore, to minimize the misunderstanding or disagreement between the coders, we designed and developed a coding guideline containing definitions of the different sentiment elements identified in the user reviews with their detailed examples. The main motive of designing and creating a coding guideline is to ensure that the two or more coders share a uniform feeling or understanding of the different sentiment elements.

S.No	Products	Product Categories	Product Sub-Categories	Review Text	Rating	Sentiment_Type
1	Watches	FastTrack	Tees Analog Watch	Waste product	1	neg
3	Watches	FastTrack	Tees Analog Watch	My product is so bad.	1	neg
1522	Clothes	Jeans	Xee' Slim Men Black	Worst quality material. Do not buy.	2	neg
2426	Shoes	TR	Derby for Man	Average Product	3	neu
2431	Shoes	TR	Derby for Man	just ok under this price	3	neu
5261	Clothes	T-shirt	Men Polo Neck White-Black	good design value for a product	4	pos
6747	Watches	Lois Caron	Analog Watch	A good one in this range FastTrack	5	pos

Fig. 2. Annotation sample of end-user comments

[3] https://drive.google.com/file/d/1F8zWWpHfXNqD1K9MOMl1XgeQjAbLdJ7z/view

To manually annotate the dataset, each coder received the developed coding guideline and the coding documents that contain the complete set of user comments and their corresponding attributes. The coding document is in Microsoft comma-separated values (CSV). An annotation sample of the crowd-user sentiments in the Amazon and Flipkart is depicted in **Fig. 2**. The annotation columns "S. No," "Product," "Product Categories," "Product Sub-categories," "Review Text," and "Rating" depict the crowd-users sentiments information in the Amazon and Flipkart shopping platforms. The column "Sentiment_Type" describes the possible sentiment type that needs to be identified by the potential annotators of the kind supporting, attacking, and neutral. The coders can select "neg" if the product review contains negative sentiments about the product under discussion. At the same time, the coder can select "neu" if the product review contains neutral sentiments or information about the product under discussion. Similarly, the coders can select "pos" if it contains positive sentiments or information about the product under discussion. Furthermore, we merge all the coding results from the three annotators to calculate the inter-coded agreement. The inter-coded agreement between the three coders was 87%, while Cohen's kappa was 68%, considered as a substantial agreement between the coders on its scale.

5.3 Data Set Annotation using Automated VADER Approach

Manually labelling customer reviews is time-consuming, laborious, and needs a lot of resources. For this purpose, we experimented with the VADER library [39], which assigns a compound score to the user comments in a range of 1 to -1. Based on the compound score, we can automatically identify each user comment's sentiments (positive, negative, or neutral) in the data set. VADER (Valence Aware Dictionary for Sentiment Reasoning) [39], introduced in 2014, is a lexicon and rule-based sentiment analysis tool used for text (review) sentiment analysis. It is sensitive to polarity (positive, negative, or neutral) and compound scores. Also, with the help of this compound score, the model decides that the text (review) is positive, negative, or neutral. The compound score is between 1 to -1. Suppose a user review is taken from the *dataset "my product is so bad"* when applying the VADER gives the result: {'neg': 0.529,'neu': 0.471,'pos': 0.0,'comp': -0.6696}. The compound polarity of the above review is -0.6696, which is negative. Therefore, the end-user review is considered a negative review. Contrary, if we annotate the dataset based on the end-user rating, there is a possibility it might lead to deceptive end-user annotation whereas, the actual end-user review represents negative sentiments towards the product while its rating represents positive sentiments. For example, an end-user review in the data set *"I can use it only for a month and after which it was damaged"* has given a rating of 5-stars. In contrast, the end-user review represents negative sentiment about the product under discussion. Therefore, building a classifier on the end-users reviews annotated on rating results in deceptive information that might lead end-users with misleading information. Thus, the VADER library might be considered an alternative source in automatically annotating the end-user reviews for sentiment identification problems. Furthermore, while manually comparing the two annotation methods (manual and automatic (VADER)), it is identified that the automatic (VADER) approach was 99% similar to the manual user comments annotation with less time and resources required as compared to the manual approach. **Table 4** shows the total number of end-user reviews categorized or classified into positive, neutral, and negative classes using the VADER library and content analysis approach. Whereas 8,645 reviews are positive, 1,663 reviews are neutral, and 1,473 reviews are negative. **Fig. 2**, shows the examples of the annotated end-user reviews using the content analysis approach and VADER.

Table 4. Total Review Categorized

Review	Number
Positive	8,645
Neutral	1,663
Negative	1,473
Total	11,781

While **Fig. 3** shows the percentage ratio of positive, neutral, and negative reviews. Approximately 70% of end-user reviews in our dataset contain positive reviews, and the rest are neutral and negative reviews.

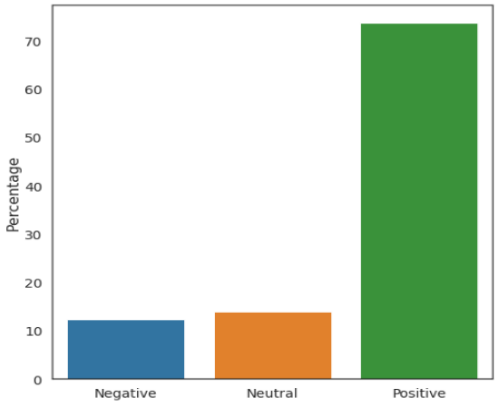

Fig. 3. Percentage of Reviews

6. Sentiment Analysis to Identify Deceptive Rating Information

Recently, researchers emphasized that users submit a large number of feedback on social media platforms that are considered a pivotal and important source of information for the software vendors [1, 23, 24]. However, manually processing and identifying helpful information for software vendors is time-consuming and challenging to recover deceptive information in social media platforms. Therefore, to automatically capture end-user opinions in the crowd-user comments on the social media platforms (Amazon & Flipkart) and identify the deceptive rating information to help end-users in decision-making, we employed five different supervised machine learning models, which are: Multinomial Naive Bayes, Logistic Regression, Gradient Boosting Classifier, Linear SVC, and Random Forest Classifier. We selected these machine learning algorithms because of their better performance and popularity on textual data classification in social media platforms [22, 23, 24]. Additionally, as depicted in **Fig. 3**, the research data set is quite imbalance; therefore, to balance the research data set, we utilized two baseline resampling approaches, such as Over-sampling and under-sampling. Also, to train and validate the machine learning classifiers, we adopted K-fold [42] and shuffled split [43] cross-validation approaches. Finally, to evaluate the performance of machine learning algorithms in identifying deceptive end-user reviews in the social media platforms, we use standard evaluation metrics, i-e, Precision, recall, and F-measure. The details about the machine learning algorithms and experiment are discussed below:

6.1 Machine Learning Algorithms

To identify deceptive end-user rating information in the Amazon and Flipkart shopping platforms, we selected various machine learning classifiers based on their better performance and accuracy on text data classification problems [22, 23, 24]. The classifiers selected are Multinomial Naive Bayes, Logistic Regression, Gradient Boosting Classifier, Linear SVC, and Random Forest Classifier. Below, we explain each machine learning classifier to give potential readers basic knowledge about the classifiers used for identifying deceptive end-user rating information on the social media platforms.

Multinomial Naïve Bayes (MNB): Thomas Bayes [25] formulated the Bayes theorem, which calculates the probability of a class occurring based on the prior knowledge of conditions related to a class. It is based on the following formula in "(1)":

$$P(A|B) = P(A) * P(B|A)/P(B) \quad (1)$$

Naïve Bayes machine learning classifier is based on Bayes theorem, and multinomial Naïve Bayes is a sub type of Naïve Bayes classifier. A Multinomial Naïve Bayes [26] classifier is a type of Naïve Bayes classifier that have been used as a baseline classifier for text classification [19]. It's a statistical classifier that plots input feature vectors to output class labels. The Multinomial Naïve Bayes combines a probability distribution of P with a fraction of documents belonging to each class, given in "(2)".

$$\Pr(c)\ \alpha\ \pi c \prod |VV|\ w{=}1\ \Pr(w|c) \quad (2)$$

Logistic Regression (LR): Logistic regression is one of the most simple machine learning techniques addressed and utilized in most data mining domains [27]. Logistic regression [28] is a second supervised machine learning method used in this research for sentiment classification. Logistic regression is a predictive analysis model that is based on probability. Logistic regression classifier grounded on a logistic function, which is also called the sigmoid function. This function maps any real value into another value between 0 and 1. This function gives an S curve, where "(3)" shows the sigmoid function.

$$f(x) = 1 / 1 + e\text{-}(x) \quad (3)$$

Where, the logistic model is defined in "(4)" [28].

$$\Pr(y/x) = \frac{1}{1+\exp(-y(\beta T\ x+\alpha))} \quad \frac{\exp(-y(\beta T\ x+\alpha))}{1+\exp(-y(\beta T\ x+\alpha))} \quad (4)$$

Gradient Boosting (GB): Friedman [29] first invented gradient boosting machines. He proposed that combining multiple simple models into one composite model yields better results instead of creating a single powerful model. Gradient boosting is one of the most powerful techniques for building predictive models. Gradient boosting is a machine learning method used for classification and regression problems, which produces a prediction model by combining the predictions of weak models, typically referred to as decision trees. Gradient Boosting Classifier supports both binary and multi Classification. Gradient Boosting Classifier depends on a loss function.

Linear Support Vector Machine (SVM): Vapnik [30] first proposed the Support Vector Machine (SVM). The SVM algorithm plots the input vector in a higher dimensional latitude or space where a consummate or maximized hyperplane separates the input data. In our case, the SVM algorithm built three parallel hyperplanes to divide the input data from each other.

The purpose of the hyperplane is to maximize the distance or length between the two parallel hyperplanes. An assumption made that the larger the distance between these parallel hyperplanes, the better the generalization error of the classifier will be [30]. SVM classifies nonlinear and linear data [10]. If the input data is linearly separable from each other, the SVM searches or finds for the linear optimal or highest separating hyperplane (the linear kernel), which is a decision boundary that separates data of one class from another. Mathematically, an unraveling or separating hyperplane can be written as W·X+b=0, where W is a weight vector of W=w1, w2,..., wn, X is a training tuple, and b is a scalar. To optimize the hyperplane, the problem essentially transforms to the minimization of ‖W‖, which is eventually computed in "(5)":

$$\sum_{i=1}^{n} \alpha i \gamma i X i \tag{5}$$

Where αi are numeric parameters, and $\boldsymbol{\gamma}$i are labels based on support vectors, Xi.

Random Forest (RF): Random forest was first proposed by Breiman [31] in 2001. The Random forest algorithm combines classification and regression tree & bagging. Furthermore, the Random Forest algorithm combines decision trees that deal with multiple numbers of parameters such as several trees to construct for the decision forest, several features to select randomly, and the depth of each tree. To make a classification decision, several decision trees need to learn. At each step of the learning method, an attribute needs to be carefully selected to split into two or more different parts. The process will continue until a pure classification split is reached. A pure classification split is when the split parts characterize only one class they belong to. In each classification split, the aim is to reach a local optimum solution [32].

6.2 Balancing Data

Balancing a dataset plays a pivotal and critical role in machine learning experiments to get stable and accurate results [33]. A balanced dataset for a machine learning classifier produces high accuracy, and the model will be trained and learned in all the classes without skewing towards the majority class. Below, we discuss imbalanced data, and we also understand and discuss about the different techniques used to balance the imbalanced datasets.

6.2.1 Imbalance Dataset

Recently, an imbalanced dataset has turned up as one of the technical challenges and problems in supervised machine learning, which might affect the performance of the supervised classifiers [34]. As shown in **Fig. 3**, the number of positive reviews in the data set is much more than neutral and negative reviews. Therefore, if we fit and train the machine learning models on imbalanced data, the model seeks an accurate performance over the majority data (positive review) while less accurate performance over the minority data (neutral and negative reviews). To handle this imbalanced dataset issue, we used the following two techniques to balance the data set, widely used in the literature to balance the data sets [34], i-e, Oversampling and Under-sampling. We use these two data balancing techniques to improve the performance of the machine learning models and predict more accurate results on minority data as compared to an imbalanced dataset.

Oversampling: Oversampling is a non-heuristic technique that balances class distribution through the random repeat of minority class examples [35]. For this purpose, we used SMOTE

[35], which produces synthetic minority examples to over-sample the minority class. Its building block is to virtually add new minority class examples by interpolating between minority class's examples that lie together. For each minority example, its k (which is set to 5 in SMOTE) nearest neighbors of the same class is considered, then some samples are randomly chosen from them according to the over-sampling rate.

Under-sampling: Under-sampling is a non-heuristic technique that targets to balance class distribution through the random exclusion or elimination of majority class samples [36]. The major disadvantage of under-sampling is that we might lose some valuable or useful data instances that might be essential for training the machine learning classifiers. For this purpose, we adopted the RandomUnderSampler [36] method for the proposed sentiment analysis approach.

Furthermore, we utilized ROC and Precision-Recall Curves to decide which data balancing (oversampling or under-sampling) approach is more suitable for the training of machine learning classifiers in our experiment. Each of them is explained in detail below:

ROC: ROC stands for Receiver Operating Characteristic curve, which encapsulates the arbitration among the false-positive-rate and true-positive-rate for a predictive model using different probability thresholds [21]. For this purpose, roc_curve ()[4] function from sklearn is used to calculate the ROC curve that takes the true outcomes (0, 1, -1) from the test dataset and the predicted probabilities for the 1 class. The function returns the true-positive-rates for each threshold, false-positive-rates for each threshold, and thresholds.

Precision-Recall Curve: Precision-Recall curves summarize the trade-off or arrangement between the true-positive rate and the positive predictive value for a model using different probability thresholds [37]. The precision_recall_curve () function from sklearn is used to calculate a precision-recall score that takes the true output values and the probabilities for the positive class as input and returns the precision, recall, and threshold values as output.

These two curves are used to evaluate or compare the model's performance using oversampling and under-sampling and decide which data balancing approach is suitable for our proposed approach.

6.3 Text Pre-processing

The data preprocessing step is comprised of the following steps:

- **Remove irrelevant characters from Review:** Firstly, we remove the irrelevant characters from the dataset, such as the punctuation, special characters, URLs, numeric values, etc., to get the significant data for training and testing the machine learning model.
- **Removal of Stop words:** Stop words are usually non-semantic words like articles, prepositions, conjunctions, and pronouns. Words such as 'the' and 'a, are articles, and pronouns such as 'it' 'we' and 'you' provide little or no information about sentiment [14]. Therefore, remove those words from the dataset to improve the training and testing efficiency of the machine learning classifiers.

[4] https://scikit-learn.org/stable/ accessed on 1-10-2021

- **Lemmatization:** Lemmatization usually refers to doing things properly using terms and morphological analysis of words, typically targeting to remove inflectional finishes only and return the base or lexicon form of a word, known as the lemma. Natural language processing toolkit used to perform lemmatization.
- **Tokenization:** It separates a string sequence into tokens such as keywords, words, symbols, phrases, and other topics called tokens. In time, tokens can be phrases, words, or entire sentences. Furthermore, some characters, such as punctuation marks, were removed in the tokenization phase [12].

6.4 Feature Extraction

Machine learning classifiers operate on numerical data. Therefore, we need methods or techniques that convert text data (user reviews) into numeric data to train and validate the machine learning models. For this purpose, we employed TF-IDF (Term Frequency Inverse Document Frequency) and CountVectorizer to convert textual data into numeric data to make the data readable for the machine learning classifiers. We selected these textual features based on their better performance on social media data [44]. Also, the only difference between TF-IDF and CountVectorizer is that TF-IDF returns the float number while the CountVectorizer returns an integer value. Each of them is elaborated on below:

6.4.1 TF-IDF

The quality of a textual sentence is depicted by the significance of words in that sentence. TF-IDF is a precise and common approach that measures and calculates how significant a word or term is in a document or review [40]. TF-IDF is an information retrieval method that contemplates a term's frequency (TF) and inverse document frequency (IDF). Since end-user comments in social media have different lengths, there is a possibility that certain words occur more frequently in lengthy user comments when compared to shorter ones. For this purpose, TF is used to balance the frequency of a word and how many times a word or term has occurred in a document or end-user review. Contrary, some words appear more frequently in a user comment, which reduces its significance referred to as stop words, such as "is," "the," "at," etc. The IDF is used to balance it by scaling up the less often used words while weighing down the most frequent words in the user reviews [15]. Also, each word or term has its TF and IDF score. Additionally, each word in the end-user review has its product score referred to as the TF*IDF weight. Furthermore, the higher the TF*IDF score (weight), the infrequent the term or word and vice-versa. Below is the equation for TF-IDF, "(6)":

$$\text{Tfidf}(t, d) = \text{tf}(t,d) * \log(N/(df+1)) \quad (6)$$

Where tf is for term frequency, t for a term (word), d for a document (review), N is the total number of documents (dataset), and df for document frequency.

6.4.2 CountVectorizer

CountVectorizer is another most frequently used approach to convert textual data to numeric data to make it parseable for the machine learning algorithms [41]. In CountVectorizer, a dictionary of all the unique words in the corpus is created and counts the number of times a term is present in the textual document or user review. The CountVectorizer uses this value as its weight. Furthermore, the CountVectorizer generates a matrix. A matrix column signifies each unique word, and a row signifies each text (review) from the document (customer review)

in the matrix. CountVectorizer counts the words in a text (review) and returns an integer value.

6.5 Training and Evaluation

6.5.1 Cross Validation

After getting the annotated data set of user comments, we applied the standard cross-validation techniques to train and validate the machine learning classifiers. For this purpose, we adopted two approaches, i-e, K-Fold, and Shuffle cross-validation, elaborated below:

- **K-Fold Cross-Validation:** In k-fold cross-validation [43], the annotated dataset of end-user comments is equally divided into k equal size of sub-datasets. In k sub-datasets, a single sub-dataset is taken as the validation data for testing the machine learning algorithm or model, and the remaining k-1 sub-datasets are used as training data [13]. The training and validation process repeats for K-times by shuffling the training and validation folds. The K-fold cross-validation approach is quite helpful to train and validate the machine learning models.
- **Shuffle split:** The shuffle split [43] cross-validation technique randomly samples the dataset during each iteration to create a training and testing set. In order words, the shuffle split divides the dataset into n splits, where each shuffle split has training and testing data.

6.5.2 Performance Measurements

To evaluate the machine learning algorithms in identifying the sentiments associated with the user comments in the social media platform, we employed the below-standard evaluation metrics.

Accuracy: Accuracy defined as "the ratio of accurately predicted reviews to total reviews", the equation of accuracy is "(7)":

$$\text{Accuracy} = \frac{\text{TP} + \text{FN}}{\text{TP+FP+FN+TF}} \tag{7}$$

Precision: Precision is defined as "the accurate predicted positive review divided by the total reviews." It describes how good a model is at predicting the positive class. The equation of precision is "(8)":

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

Recall: The recall is defined as "the accurate prediction of positive reviews divided by the actual positive class reviews", the equation of recall is "(9)":

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

F1 Score: F1 score is defined as "the weighted average of precision and recall", the equation of the F1 score is given below "(10)":

$$\text{F1 Score} = \frac{2*(\text{Recall} * \text{Precision})}{(\text{Recall} + \text{Precision})} \tag{10}$$

True-positive-rate (TPR): The true-positive-rate also called Sensitivity or Recall is defined as "the number of accurate positive predicted reviews divided by the actual positive review", the equation of TPR is the same as recall which is "(11)":

$$\text{TPR} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{11}$$

False-positive-rate (FPR): The false-positive-rate also called Specificity is defined as "the ratio of positive review which is predicted false to the total negative review", the equation of FPR is "(12)":

$$\text{TPR} = \frac{\text{FP}}{(\text{FP} + \text{TN})} \tag{12}$$

We use TPR and FPR for the ROC Curves, which are discuss and elaborated in Section 5.

6.6 Training and Evaluation

The results findings from different machine learning algorithms using the K-Fold cross-validation approach to identify the end-user opinions and recover the deceptive end-user rating information is shown in **Table 5**. The values in bold represent the highest values for each evaluation metric for their corresponding machine learning classifier. It can be seen from **Table 5**, overall, machine learning classifiers with oversampling balancing technique outer-performs the machine learning algorithms with under-sampling techniques. Similarly, the Linear SVC classifier outperforms other machine learning algorithms when identifying user opinions in the social media platforms using both Under-sampling and Over-sampling balancing approaches. **Table 5**. Shows that Linear SVC gives the maximum accuracy of 92.32% when identifying end-users sentiments in social media. Particularly the Linear SVC (BOW and SMOTE) has higher Precision, Recall, and F-measure values when identifying positive end-users opinions in the social media platform (Amazon & Flipkart), which are 93.99%, 92.32%, and 92.96%, respectively. Similarly, the Linear SVC (BOW and SMOTE) has higher Precision, Recall, and F-measure values when identifying neutral end-users opinions in the social media platform (Amazon & Flipkart), which are 93.35%, 93.49%, and 93.37%, respectively. While, the Linear SVC and RF (TF-IDF and SMOTE) have higher Precision, Recall, and F-measure values when identifying supporting end-users opinions in the social media platforms, which are 93.41%, 93.02%, and 93.16%, respectively.

Table 5. K Fold Dataset Result

Classifier	Classifier Feature	Data Balancing	K Fold = 10									
				Claims- Positive			Claims-Neutral			Claims-Negative		
			A	F1	R	P	F1	R	P	F1	R	P
MNB	MNB-TFIDF	SMOTE	83.64	85.83	83.64	90.15	85.88	83.58	90.29	86.74	86.11	88.93
	MNB-Countvectorizer	SMOTE	82.39	85.41	82.39	90.43	74.82	79.57	82.07	74.3	79.1	81.62
	MNB-TFIDF	RUS	80.78	83.85	80.78	90.41	83.85	80.78	90.41	83.85	80.78	90.41
	MNB-Countvectorizer	RUS	82.78	84.79	82.78	90.78	84.79	82.78	90.78	84.79	82.78	90.78
LR	LR-TFIDF	SMOTE	91.78	92.38	91.78	93.55	92.53	91.91	93.7	92.89	92.67	93.38
	LR-Countvectorizer	SMOTE	92.16	92.77	92.16	93.75	91.94	92.18	91.94	91.49	91.74	91.49
	LR-TFIDF	RUS	88.55	89.59	88.55	92.04	89.59	88.55	92.04	89.59	88.55	92.04
	LR-Countvectorizer	RUS	90.52	91.23	90.52	92.87	91.23	90.52	92.87	91.23	90.52	92.87
Linear SVC	Linear SVC-TFIDF	SMOTE	91.84	92.51	91.84	93.63	92.54	91.86	93.41	**93.16**	**93.02**	**93.41**
	Linear SVC-Countvectorizer	SMOTE	**92.32**	**92.96**	**92.32**	**93.99**	**93.37**	**93.49**	93.35	92.68	92.81	92.66
	Linear SVC-TFIDF	RUS	88.73	89.75	88.73	92.11	89.75	88.73	92.11	89.75	88.73	92.11
	Linear SVC-Countvectorizer	RUS	91.06	91.76	91.06	93.31	91.76	91.06	93.31	91.76	91.06	93.31
GB	GB-TFIDF	SMOTE	91.62	92.21	91.62	93.53	92.1	91.56	93.41	92.4	92.16	93.08
	GB-Countvectorizer	SMOTE	92.02	92.49	92.02	93.34	93.3	93.31	**93.43**	93.04	93.04	93.13
	GB-TFIDF	RUS	89.38	90.16	89.38	92.16	90.16	89.38	92.16	90.16	89.38	92.16
	GB-Countvectorizer	RUS	90.1	90.77	90.1	92.55	90.77	90.1	92.55	90.77	90.1	92.55
RF	RF-TFIDF	SMOTE	91.66	92.37	91.66	93.38	92.58	91.81	93.72	93.33	**93.31**	93.37
	RF-Countvectorizer	SMOTE	90.83	91.74	90.83	93.09	93.31	93.13	93.31	92.96	93.14	93.01
	RF-TFIDF	RUS	88.79	89.88	88.79	92.04	89.88	88.79	92.04	89.88	88.79	92.04
	RF-Countvectorizer	RUS	89.35	90.35	89.35	92.51	90.35	89.35	92.51	90.35	89.35	92.51

A: Accuracy, P: Precision, R: Recall, MNB: Multinomial Naïve Bayes, LR: Logistic Regression, SVC: Support Vector Classifier, GB: Gradient Boosting, RF: Random Forest, RUS: Random Under-Sampler

Also, the results finding from the different machine learning algorithms using the Shuffle Split cross-validation approach to identify the end-user opinions and recover the deceptive end-user rating information is shown in **Table 6**. The values in bold represent the highest values for each evaluation metric for their corresponding machine learning classifier. Similar to the previous experiment, **Table 6** shows that overall, machine learning classifiers with oversampling balancing technique outer-performs the machine learning classifiers with under-sampling techniques. Also, the Linear SVC classifier outperforms other machine learning algorithms when identifying user opinions in the social media platforms using both Under-sampling and Over-sampling balancing approaches with the Shuffle Split cross-validation approach. **Table 6** shows that Linear SVC with BOW and SMOTE gives the maximum

accuracy of 93.69% when identifying end-users sentiments in social media. Particularly the Linear SVC (BOW and SMOTE) has higher Precision, Recall, and F-measure values when capturing and identifying the positive, neutral, and negative end-users opinions in the social media platform (Amazon & Flipkart), which are 94.01%, 93.69%, and 93.81%, 94.11%, 93.84%, and 93.94%, 94.26%, 93.97%, and 94.08%, respectively. To summarize, the Linear SVC classifier outer performs other machine learning algorithms using both K-Fold and Shuffle Split cross-validation techniques. In particular, the **Linear SVC classifier with BOW and SMOTE** setting using Shuffle Split cross-validation technique perform better than the **Linear SVC (BOW & SMOTE) classifier** using the K-Fold cross-validation approach in terms of accuracy, precision, recall, and F-measure. Therefore, we can choose **a Linear SVC classifier with BOW and SMOTE** setting as the best classifier for our proposed approach to identifying deceptive end-user ratings in the user media platform. While obtaining the results, the SVM machine learning algorithm performs better than the other machine learning classifiers in identifying end-user sentiments to recover the deceptive rating information. The one possible reason is that a support vector machine takes the data points or output; in our case, these data points are compound scores obtained using polarity scores and draw a hyperplane that best separates the data points. This hyperplane works as a decision boundary; anything that falls to one side is a class. While the other classifiers like naïve byes and logistic regression work on probability and random forests combine the multiple decision trees. Furthermore, in the previous research, it is recovered that SVM performs better in classifying textual data from social media platforms [8, 9, 10, 12, 23, 24] that also align with our obtained research results.

Table 6. Shuffle Split Dataset Result

Classifier	Classifier Feature	Data Balancing	Shuffle Split = 10									
				Claims- Positive			Claims-Neutral			Claims-Negative		
			A	F1	R	P	F1	R	P	F1	R	P
MNB	MNB-TFIDF	SMOTE	85.92	86.59	85.92	88.81	86.47	85.88	88.64	86.74	86.11	88.93
	MNB-Countvectorizer	SMOTE	84.26	85.05	84.26	87.56	85.06	84.22	87.49	85.16	84.27	87.71
	MNB-TFIDF	RUS	83.02	84.21	83.02	88.62	84.48	83.35	88.6	84.28	83.07	88.75
	MNB-Countvectorizer	RUS	84.45	85.33	84.45	89.25	85.62	84.73	89.65	85.66	84.76	89.45
LR	LR-TFIDF	SMOTE	92.96	93.15	92.96	93.55	93.22	93.04	93.6	92.89	92.67	93.38
	LR-Countvectorizer	SMOTE	93.5	93.64	93.5	93.9	93.69	93.56	93.9	93.62	93.48	93.87
	LR-TFIDF	RUS	89.68	90.24	89.69	91.71	89.91	89.35	91.44	89.89	89.32	91.34
	LR-Countvectorizer	RUS	91.33	91.7	91.33	92.69	91.62	91.24	92.67	91.44	91.04	92.55
Linear SVC	Linear SVC-TFIDF	SMOTE	93.15	93.27	93.15	93.5	93.14	93.03	93.34	93.16	93.02	93.41
	Linear SVC-Countvectorizer	SMOTE	**93.69**	**93.81**	**93.69**	**94.01**	**93.94**	**93.84**	**94.11**	**94.08**	**93.97**	**94.26**
	Linear SVC-TFIDF	RUS	90.45	90.91	90.45	92.14	90.91	90.04	91.8	90.63	90.15	91.91
	Linear SVC-Countvectorizer	RUS	91.99	92.3	91.99	93.11	91.76	91.39	92.7	91.93	91.59	92.84
GB	GB-TFIDF	SMOTE	92.46	92.68	92.46	93.3	93.18	92.99	93.71	92.4	92.16	93.08
	GB-Countvectorizer	SMOTE	93.13	93.31	93.13	93.68	93.22	93.04	93.55	93.29	93.11	93.63
	GB-TFIDF	RUS	90.01	90.5	90.01	91.95	90.31	89.83	91.74	90.47	89.96	91.97
	GB-Countvectorizer	RUS	91.49	91.85	91.49	92.91	91.68	91.3	92.83	90.97	90.53	92.39

RF	RF-TFIDF	SMOTE	93.57	93.56	93.57	93.57	93.32	93.34	93.32	93.33	93.31	93.37
	RF-Countvectorizer	SMOTE	92.49	92.55	92.49	92.74	92.91	92.86	93.04	93.07	93.01	93.24
	RF-TFIDF	RUS	89.76	90.28	89.76	91.62	89.92	89.38	91.33	90.16	89.63	91.52
	RF-Countvectorizer	RUS	90.8	91.25	90.8	92.41	91.39	90.24	92.6	91.22	90.76	92.47

A: Accuracy, P: Precision, R: Recall, MNB: Multinomial Naïve Bayes, LR: Logistic Regression, SVC: Support Vector Classifier, GB: Gradient Boosting, RF: Random Forest, R.U.S: Random Under-Sampler

Although, we obtained comparatively better results with the traditional machine learning algorithms, such as SVM, MNB, etc. Still, we can further improve the performance of the proposed SentiDeceptive approach in recovering deceptive end-user rating information busy employing state-of-the-art deep learning and transfer learning techniques and algorithms, such as Word2Vec, Convolutional neural networks, Bert, and Sentence-Bert. Furthermore, deep learning and transfer learning algorithms perform better on larger data sets. In contrast, the research dataset curated for the SentiDeceptive approach is comparatively smaller, i-e, it contains only 11781 end-user comments from the Amazon and Flipkart platforms. For this purpose, we are interested in collecting and preparing more extensive end-user comments data set from multiple shopping platforms, such as Amazon, Flipkart, Alibaba, Shopify, and eBay, and testing the performance of state-of-the-art deep learning and transfer learning algorithms to identify deceptive end-user information.

7. SentiDeceptive Tool

In this section, we describe the steps used in the development of the Sentideceptive tool used to identify user opinions on end-user comments in the social media platforms, i-e, Amazon and Flipkart, and recover deceptive rating information to help potential customers in decision-making by selecting a specific product based on customer reviews instead of rating.

5.1 ROC Curve for best classifier

The next step is to visualize the ROC [21] curve for the best classifiers using the oversampled and under-sampled datasets to compare the most accurate classifier.

Fig. 4 shows the ROC curve for the highest accuracy and F1 score of a classifier, Linear SVC, where the green line shows the ROC curve for Oversampled data and the average AUC score for Oversampled data approximately is approximately 0.9813%. The Blue line shows the ROC curve for under-sampled data, and the average AUC score for under-sampled data is about 0.9459%. Similarly, we created a Precision-Recall curve for the different machine learning algorithms to identify the best resampling approach and best machine learning classifier with the tradeoff between the false negative and false positive. Saito and Rehmsmeier [45] highlighted that the Precision-Recall plot is more useful and informative than the ROC curve when evaluating machine learning classifiers with the imbalanced data set. Furthermore, **Fig. 4** shows the Precision-Recall curve for the highest accuracy and F1 score of a classifier, Linear SVC. The green line shows the Precision-Recall curve for Oversampled data, and the average AUC score for Oversampled data is approximately 0.8552%. The Blue line shows the Precision-Recall curve for under-sampled data, and the average AUC score for under-sampled data around is 0.8446%.

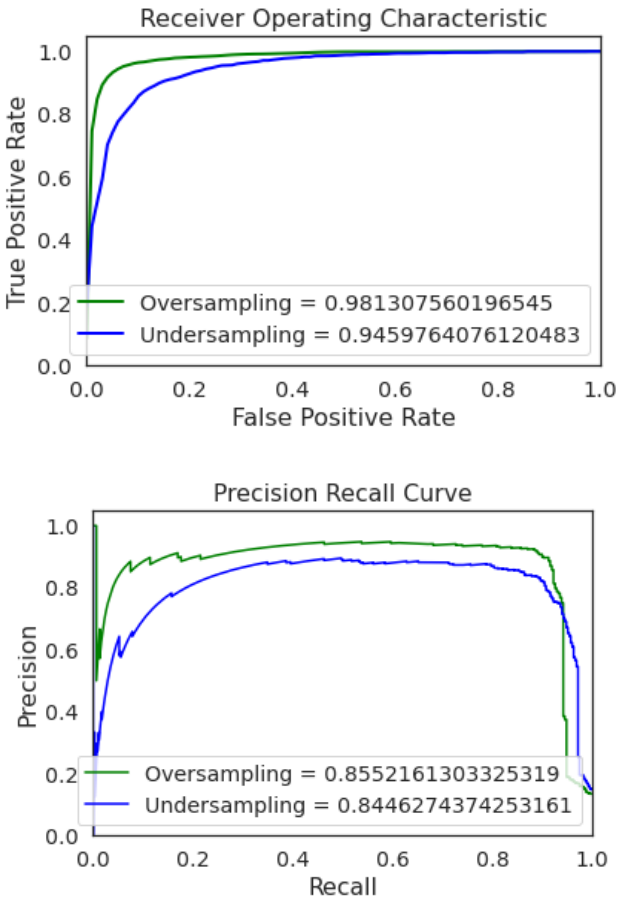

Fig. 4. ROC Curve and Precision-Recall Curve for identifying best resampling technique with SVM classifier

5.2 SentiDeceptive Tool Environment

Next, to develop the SentiDecptive tool, we choose to build a web-based application using HTML, CSS, and Bootstrap as the front-end development and Python Django framework as a back-end development tool. Furthermore, we used the MySQL database as a tool to store the end-users reviews from the social media platforms.

5.3 Implementing best Machine Learning Algorithm

Next, we need to integrate the best machine learning classifier with the web-based application, i-e, Linear SVM, selected based on the results obtained from the machine learning experiment, as explained in sub-section 6.6. For this purpose, we used the Pickle library, which saves the machine learning model as “model. p”. Later, we use this file in the web-based application to predict the output of any newly submitted end-user review in the web application.

5.4 Review Classification Process of SentiDeceptive Tool

The SentiDeceptive works by integrating the saved model (model. p) with the Python framework (Django), used as a front-end tool. After combining the model, the user will predict the required output of the end-user reviews, i-e, positive, negative, or neutral. For this purpose,

the end-users will submit their corresponding review about the specific product in the web-based application, and that end-user review will be saved in the database for future use. Furthermore, the end-users review will be processed by the trained machine learning algorithm intertwined with the web-based application to get the desired output and update the Pi chart of that specific product in the web application. **Fig. 5** shows an excerpt of SentiDeceptive tool, developed in python Django framework[5]. The working of the SentiDeceptive tool with the end-user comments is elaborated as: in the first step, the end-user writes a review about a specific product (Bata, Adidas, etc.) in the SentiDeceptive tool. In step 2, the end-user will submit the review in the SentiDeceptive tool. After submitting, the trained machine learning model will process the input end-user feedback and predict the desired classification label, i-e, positive, neutral, or neutral. In step 3, the pi diagram will be updated concerning the review classified in the previous step. In step 4, the product's average rating under discussion is displayed in the SentiDeceptive tool, which is calculated based on the identified end-user sentiments using a machine learning algorithm. Finally, in step 5, the end-user review will be displayed in the SentiDeceptive tool along with the identified sentiment class (positive, neutral, and negative).

Furthermore, we manually compare the sentiments type (supporting, attacking, or neutral) predicted by the SentiDeceptive tool with the exiting end-user labelled instances in the data set. We found that the SentiDeceptive tool predicts 98 end-user comments as correctly classified into the type of sentiment supporting, attacking, and neutral out of 100 crowd-user comments. Therefore, we can conclude that the SentiDeceptive tool's accuracy is defined to be 98%. However, it still needs to be tested thoroughly and comprehensively with more significant data instances from the data set to ensure its correctness and effectiveness in identifying deceptive end-user rating information and improving end-user buying or purchasing decision-making. Additionally, the proposed SentiDeceptive approach lacks a pivotal aspect, i-e, an evaluation of the proposed SentiDeceptive tool and its comparison to similar tools. The possible reason is that according to our knowledge, we didn't find any relevant and similar tool to compare the proposed SentiDeceptive approach, still date. Moreover, in future work, we need to focus on this vital aspect and conduct a series of experiments with the students and software professionals about the authenticity of the results obtained with the proposed SentiDeceptive tool.

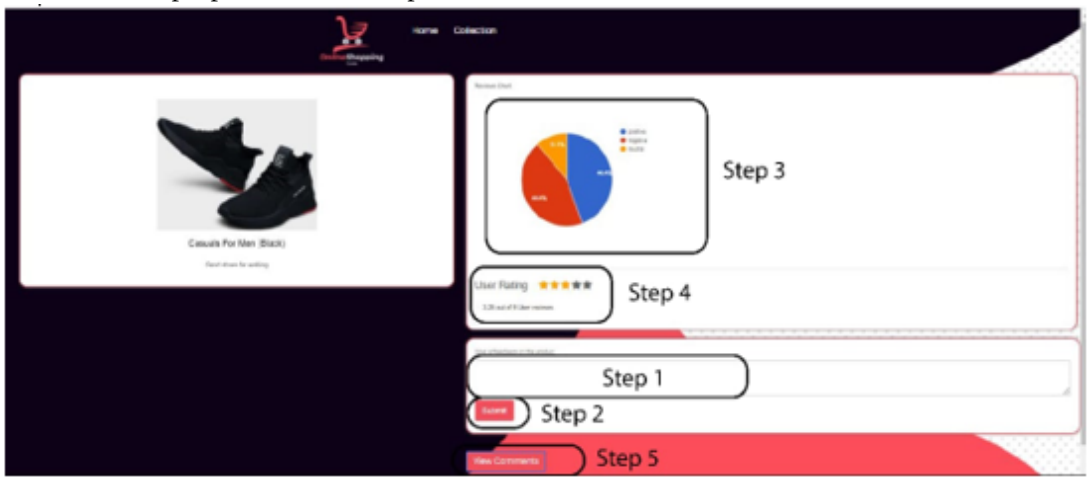

Fig. 5. Screenshot of Item Collection

[5] https://www.djangoproject.com/ Accessed on 12-6-2021

8. Identifying Deceptive rating information with SentiDeceptive approach

The proposed SentiDeceptive approach identifies deceptive rating information in the Amazon and Flipkart shopping platforms by processing actual end-user textual comment information instead of crowd-user ratings. The motivation for identifying deceptive rating information is while processing end-user comments in the data set collected from the Amazon and Flipkart shopping platforms. We found that end-user reviews, where text represents the negative information, but the rating is 5-stars, i.e., *"very bad use chattering work dress", "the worst product ever it is not separate it is attached with it you cannot wear the only shirt it is very bad"*. At the same time, end-user reviews play a pivotal role in purchasing or buying a specific product. However, the truthfulness and authenticity of these users' reviews are still not guaranteed. These reviews, news websites, and content platforms are susceptible to deceptive information [3]. A recent study reported that 87% of customers or end-users change their purchase or buying decisions after reading positive feedback. At the same time, 80% of customers change their purchasing decision after confronting or reading negative reviews [5]. Also, the customer accuracy in identifying deceptive opinions in online user reviews is 61.9 % [6]. For this purpose, our proposed SentiDeceptive approach analyzes each end-user comment in the data using natural language processing and a machine learning classifier set first to identify its sentiment, i-e, positive, negative, or neutral. Next, after identifying the sentiment of the end-user comment in the social media platform, the Pi diagram is updated in response to the identified sentiment type, thus overcoming deceptive rating information. Furthermore, to support our claim, we developed a tool in Python that implements the SentiDeceptive approach, as explained in section 7. In a nutshell, the proposed SentiDeceptive approach help end-users in providing provide accurate and reliable information about the various products in the online shopping platforms by summarizing and aggregating the overall end-user opinions.

9. Conclusion and Future directions

In this paper, we proposed a SentiDeceptive approach, which automatically classifies end-user reviews into negative, positive, and neutral sentiments and identifies online product ratings based on crowd-users comments in social media to recover deceptive end-user rating information. For this purpose, we first collected 11781 end-users comments from the Amazon store and Flipkart web application covering distant products, such as watches, mobile, shoes, clothes, and perfumes. Next, we develop a coding guideline used as a base for the comments annotation process. We then applied the content analysis approach and VADER library to annotate the end-user comments in the data set with the identified codes, which results in a labelled data set used as an input to the machine learning classifiers. We conclude that the VADER library can be used as an alternative source for automatically annotating end-user comments in the social media platform for sentiment classification to overcome the hectic and time-consuming manual annotation process with better results and accuracy. Finally, we applied the sentiment analysis approach to identify the end-users opinions and overcome the deceptive rating information in the social media platforms by first preprocessing the input data to remove the irrelevant (stop words, special characters, etc.) data from the dataset. Next, to identify reliable results with the machine learning algorithms, we employ two standard resampling approaches to balance the data set, i-e, oversampling, and under-sampling. It is determined that machine learning algorithms with oversampling resampling approach perform better as compared to the under-sampling method. Next, we extract different features (TF-IDF

and BOW) from the textual data in the data set and then train & test the machine learning algorithms by applying a standard cross-validation approach (KFold and Shuffle Split). It is concluded with the machine learning experiment that Shuffle Split cross-validation approach outer perform Kfold validation approach. Also, we concluded from the two machine learning experiments that **the Linear SVC classifier with BOW and SMOTE** setting as the best classifier for the proposed approach to identifying deceptive end-user ratings in the user media platform by developing a SentiDeceptive methodology. The proposed method also proves to be a better classification tool for processing end-user comments from multiple sources, i-e, Amazon and Flipkart. However, we are interested in including end-user feedback from other shopping platforms, such as Alibaba, Shopify, and eBay, in testing the performance of the SentiDecptive approach.

A large amount of end-user comments in the various social media platforms, i-e, amazon store, Flipkart, app stores, Twitter, and other e-commerce websites, provide an essential source of information for the vendors' organizations, software developers, and end-users to identify, captures, and analyze diverse user opinions in these freely available crowd-users discussions. Also, it helps potential users and vendor organizations refine certain decision-making to improve their businesses and purchase decisions by incorporating end-user feedback from social media platforms. However, it put numerous challenges on the end-users to quickly and efficiently sort out exiting user conversations in the social media platforms and accommodate their reviews in the ongoing process. Also, the sentiment information found in the social media platforms, i-e, Amazon, Flipkart, etc., are calculated based on the end-user rating that might be deceptive, leading potential users to miss information and products. To remedy this and facilitate end-users, we proposed a SentiDeceptive approach that summarises overall end-user opinions about the product under discussion in the social media platform by analyzing end-user comments to provide accurate and reliable products information. Additionally, to help end-users in decision-making, we employ argumentation theory [46, 47], rationale mining [24, 56], and recommendation approaches [61, 62, 63] together with the SentiDeceptive approach to identify alternative products or features by analyzing end-user feedback in the social media platform. Besides, contingency tables or mosaic plots can give an overall understanding of the products under discussion to the incoming end-users by highlighting important information, such as alternative features, issues, supporting arguments, attacking arguments.

The dataset used in this research is collected from the Amazon and Flipkart e-commerce platforms, where the specified end-users participated in curating this dataset. It may lead to customer opinion threats. It can be overcome by looking into more data sources from different e-commerce platforms, i-e, Alibaba, Shopify, and eBay, to enhance the dataset size and further improve the performance of our proposed methodology. Although we get encouraging results with the machine learning classifiers, still, we are interested in expediting the existing state-of-the-art deep learning classifiers to improve the accuracy, precision, recall, and f-measure of the deep learning classifiers. Another possible threat, authors of the research approach who took part in the annotation of the end-user comments were also involved in the design and implementation of the classification experiment. However, the manuscript authors performed the coding and annotation process in a professional, iterative, and systematic way; there is still a probability that the manuscript authors have subconsciously bid for a second guess. Furthermore, the proposed approach lacks a pivotal aspect, i-e, an evaluation of the proposed SentiDeceptive tool and its comparison to similar tools. Additionally, we might have missed some significant and promising machine learning features that could enhance and improve the performance of machine learning algorithms in identifying the deceptive end-user rating information in the social media platform.

In the future, we are interested in compressively analyzing and evaluating the proposed SentiDeceptive tool with some real-time case studies and comparing its performance to similar machine learning tools. For this purpose, we are interested in employing the proposed SentiDeceptive approach in the IoT domain [48, 49, 50]. Another possible future direction is to use argumentation mining [46] to automatically identify the relationship between the two user comments [47] in the social media platform and remove the overhead of the annotation process. Furthermore, in the future, we aim to use the state-of-the-art deep learning classifiers on a relatively larger data set by collecting more data instances from different e-commerce platforms such as Alibaba, Shopify, and eBay, to further improve the performance of the SentiDeceptive tool. Moreover, in future work, we need to focus on this vital aspect and conduct a series of experiments with the students and software professionals about the authenticity of the results obtained with the proposed SentiDeceptive tool.

Acknowledgement

Taif University Researchers Supporting Project number (TURSP-2020/126), Taif University, Taif, Saudi Arabia

References

[1] D. Pagano, and W. Maalej, "User feedback in the appstore: An empirical study," in *Proc. of 2013 21st IEEE international requirements engineering conference (RE)*, Rio de Janeiro, Brazil, pp. 125-134, 15 July 2013. Article (CrossRef Link)

[2] W. P. Risk, G. S. Kino, and H. J. Shaw, “Fiber-optic frequency shifter using a surface acoustic wave incident at an oblique angle,” *Opt. Lett.*, vol. 11, no. 2, pp. 115–117, 1986. Article (CrossRef Link)

[3] O. Cocarascu and F. Toni, "Detecting deceptive reviews using argumentation," in *Proc. of the 1st International Workshop on AI for Privacy and Security*, Imperial College London, pp. 1-8, 2016. Article (CrossRef Link)

[4] R. Abinaya., P. Aishwaryaa, S. Baavana, and N. T. Selvi, "Automatic sentiment analysis of user reviews," in *Proc. of 2016 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR)*, Chennai, India, pp. 158-162, 16 July 2016. Article (CrossRef Link)

[5] D.H. Fusilier, M. Montes-y-Gómez, P. Rosso, and R. G. Cabrera, "Detecting positive and negative deceptive opinions using PU-learning," *Information processing & management*, vol. 51, no. 4, pp. 433-443, 2015. Article(CrossRef Link)

[6] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," *arXiv preprint arXiv: 1107. 4557*, 2011. Article (CrossRef Link)

[7] B. Pang, and L. Lee, “Opinion mining and sentiment analysis,” *Foundations and trends in information retrieval*, vol. 2, no. 1-2, pp. 1-135, 07 Jul 2008. Article (CrossRef Link)

[8] Z. Singla, S. Randhawa, and S Jain, "Sentiment analysis of customer product reviews using machine learning," in *Proc. of 2017 international conference on intelligent computing and control (I2C2)*, Coimbatore, India, pp. 1-5, 2017. Article (CrossRef Link)

[9] S. Dey, S. Wasif, D. S. Tonmoy, S. Sultana, J. Sarkar, and M. Dey, "A comparative study of support vector machine and Naive Bayes classifier for sentiment analysis on Amazon product reviews," in *Proc. of 2020 International Conference on Contemporary Computing and Applications (IC3A)*, Lucknow, India, pp. 217-220, 2020. Article (CrossRef Link)

[10] T. U. Haque, N. N. Saber, and F. M. Shah, "Sentiment analysis on large scale Amazon product reviews," in *Proc. of 2018 IEEE international conference on innovative research and development (ICIRD)*, Bangkok, Thailand, pp. 1-6, 2018. Article (CrossRef Link)

[11] C. Rain, "Sentiment Analysis in Amazon Reviews Using Probabilistic Machine Learning," M.S thesis, Department of Computer Science, Swarthmore College, Swarthmore, PA, USA, 2013. Article (CrossRef Link)
[12] S. S. Sikarwar, Dr. N. Tiwari, "Analysis The Sentiments Of Amazon Reviews Dataset By Using Linear SVC And Voting Classifier," *International journal of science and technology research*, vol. 9, no. 6, pp. 461-465, 2020. Article (CrossRef Link)
[13] M. S. Lakshmi, S. P. Kumar, M. Janardhan, "Machine Learning Centric Product Endorsement on Flipkart Database," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 8, no. 6, pp. 2750-2753, 2019. Article (CrossRef Link)
[14] P. V. Rajeev, V. S. Rekha, "Recommending Products to Customers using Opinion Mining of Online Product Reviews and Features," in *Proc. of 2015 International Conference on Circuit, Power and Computing Technologies(ICCPCT)*, Nagercoil, India, 2015. Article (CrossRef Link)
[15] G. Kaur, and A. Singla, "Sentimental analysis of Flipkart reviews using Naïve Bayes and decision tree algorithm," *International Journal of Advanced Research in Computer Engineering & Technology*, vol. 5, no. 1, 2016.
[16] M. Yasen, and S. Tedmori, "Movies reviews sentiment analysis and classification," in *Proc. of 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, Amman, Jordan, pp. 860-865, 2019. Article (CrossRef Link)
[17] S. Ahmed, and A. Danti, "A novel approach for Sentimental Analysis and Opinion Mining based on SentiWordNet using web data," in *Proc. of 2015 International Conference on Trends in Automation, Communications and Computing Technology (I-TACT-15)*, Bangalore, India, pp. 1-5, 2016. Article (CrossRef Link)
[18] E. Boiy, P. Hens, K. Deschacht, and M. F. Moens, "Automatic Sentiment Analysis in On-line Text," in *Proc. of ELPUB*, Leuven, Belgium, pp. 349-360, 2007. Article (CrossRef Link)
[19] J. Corbin and A. Strauss, *Basics of qualitative research: Techniques and procedures for developing grounded theory*, 4th ed. California, USA: Sage publications, 2014. Article (CrossRef Link)
[20] K. A. Neuendorf, *The content analysis guidebook*, 2nd Ed., California, USA: Sage publications, 2017. Article (CrossRef Link)
[21] J. A. Hanley, and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve," *Radiology*, vol.143, no. 1, 1982. Article (CrossRef Link)
[22] J. Hartmann, J. Huppertz, C. Schamp, and M. Heitmann, "Comparing automated text classification methods," *International Journal of Research in Marketing*, vol. 36, no. 1, pp. 20-38, 2018. Article (CrossRef Link)
[23] J. A. Khan, L. Liu, L. Wen, and R. Ali, "Conceptualising, extracting and analysing requirements arguments in users' forums: The CrowdRE-Arg framework," *Journal of Software: Evolution and Process*, vol. 32, no. 12, pp. 1-34, 2020. Article (CrossRef Link)
[24] J. A. Khan, L. Liu, and L. Wen, "Requirements knowledge acquisition from online user forums," *IET Software*, vol. 14, no. 3, pp. 242-253, 2020. Article (CrossRef Link)
[25] D. D. Lewis, "Naive (Bayes) at forty: The independence assumption in information retrieval," in *Proc. of European conference on machine learning*, Berlin, Heidelberg, pp. 4-15, 1998. Artitcle (CrossRef Link)
[26] M. Abbas, K. A. Memon, A. A. Jamali, S. Memon, and A. Ahmed, "Multinomial Naive Bayes classification model for sentiment analysis," *Int. J. Comput. Sci. Netw. Secur*, vol. 19, no. 3, 2019. Article (CrossRef Link)
[27] D. R. Cox, "The regression analysis of binary sequences," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 20, no. 2, pp. 215-232, 1958. Article (CrossRef Link)
[28] Jr. D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied logistic regression*, Vol. 398, Hoboken, New Jersey, USA: John Wiley & Sons, 2013. Article (CrossRef Link)
[29] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, vol. 29, no. 5, pp. 1189-1232, 2001. Article (CrossRef Link)
[30] C. Cortes, and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273-297, 1995. Article (CrossRef Link)

[31] T. K. Ho, "Random decision forests," in *Proc. of 3rd international conference on document analysis and recognition*, Montreal QC, Canada, pp. 278-282, 2002. Article (CrossRef Link)
[32] B. Vamsi, N. Suneetha, Ch. Sudhakar and K. Amaravati, "Sentiment Analysis on Online Reviews using Supervised Learning: A Survey," *International Journal of Control Theory and Applications*, vol. 10, no. 30, pp. 143-152, 2017. Article (CrossRef Link)
[33] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 1-5, 2017. Article (CrossRef Link)
[34] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Special issue on learning from imbalanced data sets," *ACM SIGKDD explorations newsletter*, vol. 6, no. 1, pp. 1-6, 2004. Article (CrossRef Link)
[35] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321-357, 2002. Article (CrossRef Link)
[36] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Handling imbalanced datasets: A review," *GESTS International Transactions on Computer Science and Engineering*, vol. 30, no. 1, pp.25-36, 2005. Article (CrossRef Link)
[37] J. Keilwagen, I. Grosse, and J. Grau, "Area under precision-recall curves for weighted and unweighted data," *PloS one*, vol. 9, no. 3, pp. e92209, 2014. Article (CrossRef Link)
[38] A. Agarwal, Bi. Xie, I. Vovsha, O. Rambow, and R. J. Passonneau, "Sentiment analysis of twitter data," in *Proc. of the workshop on language in social media (LSM 2011)*, Portland, Oregon, USA, pp. 30-38, 2011. Article (CrossRef Link)
[39] C. Hutto, and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Proc. of the International AAAI Conference on Web and Social Media*, Michigan, USA, pp. 1-10, 2015. Article (CrossRef Link)
[40] T. Joachims, "A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization," *Carnegie-mellon univ pittsburgh pa dept of computer science*, 01 March 1996. Article (CrossRef Link)
[41] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," *Expert Systems with Applications*, vol. 57, no. 15, pp. 117-126, 2016. Article (CrossRef Link)
[42] J. D. Rodriguez, A. Perez, and J. A. Lozano, "Sensitivity analysis of k-fold cross validation in prediction error estimation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 3, pp. 569-575, 2009. Article (CrossRef Link)
[43] R. M. Czekster, P. Fernandes, J. M. Vincent, and T. Webber, "Split: a flexible and efficient algorithm to vector-descriptor product," in *Proc. of VALUETOOLS*, p. 83. 2007. Article (CrossRef Link)
[44] D. D. Lewis, "Feature selection and feature extraction for text categorization," in *Proc. of Speech and Natural Language: Proceedings of a Workshop Held at Harriman*, New York, pp. 212-217, February 1992. Article (CrossRef Link)
[45] T. Saito, and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," *PloS one*, vol. 10, no. 3, no. 4, pp. e011843, 2015. Article (CrossRef Link)
[46] K. Atkinson, P. Baroni, M. Giacomin, A. Hunter, H. Prakken, C. Reed, G. Simari, M. Thimm, S. Cillata, "Towards Artificial Argumentation," *AI Magazine*, vol. 38, no. 3, pp. 25-36, 2017. Article (CrossRef Link)
[47] M. Lippi, and P. Torroni, "Argumentation mining: State of the art and emerging trends," *ACM Transactions on Internet Technology*, vol. 16, no. 2, pp. 1-25, 2016. Article (CrossRef Link)
[48] M. D. Alshehri, F. Hussain, M. Elkhodr, B. S. Alsinglawi, "A distributed trust management model for the internet of things (DTM-IoT)," *Recent Trends and Advances in Wireless and IoT-enabled Networks*, Springer, Cham, pp. 1-9, 2019. Article (CrossRef Link)
[49] M. D. Alshehri, F. K. Hussain, "A fuzzy security protocol for trust management in the internet of things (Fuzzy-IoT)," *Computing*, vol. 101, no. 7, pp. 791-818, 2019. Article (CrossRef Link)

[50] M. D. Alshehri, F. K. Hussain, O. K. Hussain, "Clustering-driven intelligent trust management methodology for the internet of things (CITM-IoT)," *Mobile networks and applications*, vol. 23, no. 3, pp. 419-431, 2018. Article (CrossRef Link)
[51] M. D. Alshehri, F. K. Hussain, "A centralized trust management mechanism for the internet of things (ctm-iot)," in *Proc. of International conference on broadband and wireless computing, communication and applications*, Fukuoka, Japan, pp. 533-543, 2017. Article (CrossRef Link)
[52] M. Elkhodr, B. Alsinglawi, M. Alshehri, "Data provenance in the internet of things," in *Proc. of 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Krakow, Poland, pp. 727-731, 2018. Article (CrossRef Link)
[53] M. Elkhodr, B. Alsinglawi, M. Alshehri, "A privacy risk assessment for the Internet of Things in healthcare," *Applications of intelligent technologies in healthcare*, Springer Nature Switzerland, Springer, Cham, pp. 47-54, 2019. Article (CrossRef Link)
[54] M. D. Alshehri, F. K. Hussain, "A comparative analysis of scalable and context-aware trust management approaches for internet of things," in *Proc. of International conference on neural information processing*, Sanur, Bali, Indonesia, pp. 596-605, 2015. Article (CrossRef Link)
[55] J. A. Khan, L. Liu, L. Wen, and A. Raian, "Crowd Intelligence in Requirements Engineering: Current Status and Future Directions," in *Proc. of Int. Conf. Requirements Engineering: Foundation for Software quality*, Essen, Germany, pp 245–261, 2019. Article (CrossRef Link)
[56] J. A. Khan, Y. Xie, L. Liu, L. Wen, "Analysis of requirements-related arguments in user forums," in *Proc. of the IEEE International Conference on Requirements Engineering*, Jeju, South Korea, pp. 63–74, 2019. Article (CrossRef Link)
[57] A. S. M. Al-Qahtani, "Product Sentiment Analysis for Amazon Reviews," *International Journal of Computer Science & Information Technology (IJCSIT)*, vol. 13, no. 3, 2021. Article (CrossRef Link)
[58] A. Shah, "Sentiment Analysis of Product Reviews Using Supervised Learning," *Reliability: Theory & Applications*, vol. 16, no. S1, pp. 243-253, 2021. Article (CrossRef Link)
[59] P. Nandwani, and R. Verma, "A review on sentiment analysis and emotion detection from text," *Social Network Analysis and Mining*, vol. 11, no. 1, pp. 1-19, 2021. Article (CrossRef Link)
[60] S. Hu, A. Kumar, F. Al-Turjman, S. Gupta, and S. Seth, "Reviewer credibility and sentiment analysis based user profile modelling for online product recommendation," *IEEE Access*, vol. 8, pp. 26172-26189, 2020. Article (CrossRef Link)
[61] G. Zhao, P. Lou, X. Qian, and X. Hou, "Personalized location recommendation by fusing sentimental and spatial context," *Knowledge-Based Systems*, vol. 196, pp. 1-16, 2020. Article (CrossRef Link)
[62] G. Zhao, Z. Liu, Y. Chao, and X. Qian, "CAPER: Context-aware personalized emoji recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 9, pp. 3160-3172, 2021. Article (CrossRef Link)
[63] G. Zhao, X. Lei, X. Qian, and T. Mei, "Exploring users' internal influence from reviews for social recommendation," *IEEE transactions on multimedia*, vol. 21, no. 3, pp. 771-781, 2018. Article (CrossRef Link)

ENGR.MUHAMMAD IRFAN is working as a Lecturer in Department of Software Engineering, University of Science and Technology Bannu, KPK, Pakistan Since April 2015. He did MSc (18Y) in Software Engineering from University of Engineering and Technology Peshawar, Pakistan in 2018 with Major in Semantic Web. He did BSc in Computer Software Engineering from University of Engineering and Technology Peshawar, Pakistan in 2009.His areas of interest is Semantic Web, Machine/ Deep Learning and Internet of Things(IoT.

Javed Ali Khan is working as an Assistant Professor in Department of Software Engineering, University of Science and Technology Bannu, Pakistan Since December 2011. He completed his PhD in Software Engineering from Tsinghua University, China. He did MSc in Software Engineering from the Baheria University Islamabad, Pakistan in 2013 and BSc in Computer Software Engineering from University of Engineering and Technology Peshawar, Pakistan in 2009. He has published more than 15 papers in reputable journals and conferences in requirements and software engineering. His areas of interest are Requirements Engineering, CrowdRE, Argumentation and argument mining, Feedback Analysis, Empirical Software Engineering, Sentiment Analysis and opinion mining.

Dr. Mohammad Dahman Alshehri is an Assistant Professor at the Computer Science Department, Taif University, Saudi Arabia and Visiting Professor at the School of Computer Science at the University of Technology Sydney (UTS), Australia. He received his PhD in Artificial Intelligence (AI) in Cybersecurity for Internet of Things (IoT) from the University of Technology Sydney, Australia. He is an IEEE Senior Member, Fellow of the Higher Education Academy (FHEA) from the UK and Ambassador of Future Technologies at Stanford University, USA. He developed several smart novel algorithms to reinforce AI-based cybersecurity for IoT to detect various cyber-attacks and provide full security and protection platform for the IoT from the most serious cyber-attacks. Furthermore, he was granted 2 patents of invention AI in Cybersecurity for IoT, also he published several publications in high ranked international journals, top-tier conferences and chapters of books, moreover, he received a number of international and national awards and prizes. His main current research interest lies in the areas of Cybersecurity, Artificial Intelligence (AI), Internet of Things (IoT).

Muhammad Asghar Ali received the B.Sc. degree in Software engineering from the University of Science and Technology Bannu, Pakistan. His research interests are within the areas of sentiment analysis, machine learning, software engineering.

Hizbullah received the B.Sc. degree in Software engineering from the University of Science and Technology Bannu, Pakistan. His research interests are within the areas of software engineering, web development, and deep learning.

Haider Ali received the B.Sc. degree in Software engineering from the University of Science and Technology Bannu, Pakistan. His research interests are within the areas of opinion mining and feedbak analysis, machine learning, requirements engineering.

ENGR.MUHAMMAD ASSAM is working as a Lecturer (On study Leave) in Department of Software Engineering, University of Science and Technology Bannu, KPK, Pakistan Since November 2011. He is currently pursuing Ph.D. in Computer Science and Technology from Zhejiang University; PR China. He did MSc (18Y) in Software Engineering from University of Engineering and Technology Taxila, Pakistan in 2018. He did BSc in Computer Software Engineering from University of Engineering and Technology Peshawar, Pakistan in 2011.His areas of interest are Brain Machine Interface, Medical Image Processing, Machine/ Deep Learning, Internet of Things (IoT) and Computer Vision.