

Article

Methodology for Analyzing the Traditional Algorithms Performance of User Reviews Using Machine Learning Techniques

Abdul Karim ¹, Azhari Azhari ¹, Samir Brahim Belhaouri ^{2,*}, Ali Adil Qureshi ³ and Maqsood Ahmad ³

¹ Department of Computer Science and Electronics, University Gadjah Mada, Yogyakarta 55281, Indonesia; abdulkarim@mail.ugm.ac.id (A.K.); arisn@ugm.ac.id (A.A.)

² Division of Information & Computer Technology, College of Science & Engineering, Hamad Bin Khalifa University, P.O. Box 5825, Doha, Qatar

³ Department of Computer Science, Khwaja Fareed University of Engineering & Information Technology, Rahim Yar Khan 64200, Pakistan; shah87061@gmail.com (A.A.Q.); maqsoodzee@gmail.com (M.A.)

* Correspondence: sbelhaouari@hbku.edu.qa

Received: 12 May 2020; Accepted: 2 July 2020; Published: 18 August 2020



Abstract: Android-based applications are widely used by almost everyone around the globe. Due to the availability of the Internet almost everywhere at no charge, almost half of the globe is engaged with social networking, social media surfing, messaging, browsing and plugins. In the Google Play Store, which is one of the most popular Internet application stores, users are encouraged to download thousands of applications and various types of software. In this research study, we have scraped thousands of user reviews and the ratings of different applications. We scraped 148 application reviews from 14 different categories. A total of 506,259 reviews were accumulated and assessed. Based on the semantics of reviews of the applications, the results of the reviews were classified negative, positive or neutral. In this research, different machine-learning algorithms such as logistic regression, random forest and naïve Bayes were tuned and tested. We also evaluated the outcome of term frequency (TF) and inverse document frequency (IDF), measured different parameters such as accuracy, precision, recall and F1 score (F1) and present the results in the form of a bar graph. In conclusion, we compared the outcome of each algorithm and found that logistic regression is one of the best algorithms for the review-analysis of the Google Play Store from an accuracy perspective. Furthermore, we were able to prove and demonstrate that logistic regression is better in terms of speed, rate of accuracy, recall and F1 perspective. This conclusion was achieved after preprocessing a number of data values from these data sets.

Keywords: machine learning; preprocessing; semantic analysis; text mining; term frequency/inverse document frequency (TF/IDF); scraping; Google Play Store

1. Introduction

In an information era where a large amount of data needs to be processed every day, minute and second—and the huge demand on computers with high processing speeds to outcome accurate results within nanoseconds, it is said that all approximately 2.5 quintillion bytes of data can be manually or automatically generated on a daily basis using different tools and application. Moreover, this illustrates the importance of text-mining techniques in handling and classifying data in a meaningful way. There are a variety of applications that help in classifying a string or a text, such as those used to detect user sentiments on comments or tweets and classification of an e-mail as spam. In this research,

we were able to categorize data based on a given text and provide some relevant information about the category accordingly using the essential and vital tasks of natural language processing (NLP) [1].

Mobile application stores enable users to search, buy and install mobile-related apps and allow them to add their comments in the form of evaluation and reviews. Such reviews and mobile program ecosystem have plenty of information regarding the user's expectations and experience. Moreover, here occurs the importance of the role played by programmers and application store regulators who can leverage the data to better understand the audience, their needs and requirements. Browsing mobile application stores show that hundred and thousands of apps are available and implemented every day which makes it necessary to perform aggregation studies using data mining techniques in which some academic studies focuses on user testimonials and mobile program stores, in addition to studies analyzing online product reviews. In this article, we used various algorithms and text classification techniques using Android application reviews [2].

App stores—or application distribution platforms—allow consumers to search, buy and set up applications. These platforms also enable users to discuss their opinions about an application in text testimonials, where they can, e.g., highlight a good feature in a specific application or request for a new feature [3]. Recent studies have revealed that program shop reviews include useful information for analysts. This feedback represents the "voice of the users" and can be employed to drive the growth effort and enhance upcoming releases of the application [4]. In this research, we cover the below main points:

1. We scraped recent Android application reviews by using the scraping technique, i.e., BeautifulSoup 4 (bs4), request, regular expression(re);
2. We scraped raw data from the Google Play Store, collected these data in chunks and normalized the dataset for our analysis;
3. We compared the accuracy of various machine-learning algorithms and found the best algorithm according to the results;
4. Algorithms can check the polarity of sentiment based on whether a review is positive, negative or neutral. We can also prove this using the word cloud corpus.

This research serves our key contributions as below:

- One of the key findings is that logistic regression performs better compared to random forest and naïve Bayes multinomial to multi-class data;
- A good preprocessing affects the performance of machine learning models;
- Term frequency (TF) overall results after preprocessing were better than the term frequency/inverse document frequency (TF/IDF).

Text mining—also referred to as text data—is the process for deriving data in which information is derived via patterns inventing of and trends along with pattern learning [5]. Text mining requires simplifying the input text through parsing—typically, along with the accession of many derived linguistic aspects and the removal of others—and the following insertion into a database, even deriving patterns inside the structured information, and last, interpreting and analyzing the output [6]. Text mining describes a combo of meanings that is novelty and fascination. Typical text mining jobs comprise text categorization, text clustering, concept/entity extraction, production of granular taxonomies, opinion analysis, document summarization and connection mimicking, i.e., understanding links between named entities [7].

Several constraints prevent analysts and development teams from utilizing the information in the reviews. To explain this further, user reviews, which demand considerable effort are available on program stores. Recent research showed that iOS consumers submit around 22 reviews per day per application [8].

Top-rated apps, like Facebook, get more than 4000 reviews. Second, the quality of the reviews fluctuates widely, from helpful advice to sardonic comments. Third, a review is nebulous, which

makes it challenging to filter negative from positive feedback. In addition to that, the star rating of a certain application represents the mean of the overall reviews done to be the users combining positive and negative ratings and hence is limited for the application development group [9].

In linguistics, evaluation is the process of relating syntactic constructions, in the degree of words, phrases, sentences and paragraph composing. In addition, it entails removing characteristics specific to the extent that this type of project is possible, to contexts [10]. The components of idiom and figurative speech, being cultural, are often additionally converted into invariant significance in semantic evaluation. Semantics—although associated with pragmatics—is different in that the former deals using term- or sentence-choice in any given circumstance, while pragmatics consider the meaning that is exceptionally derived from tone or context. In other words, in various conditions, semantics is about the meaning and pragmatics, the meaning encoded keywords which are translated by an audience [11]. In information retrieval, TF/IDF, brief for term frequency/inverse document frequency, is a statistic that is meant to reveal how important a word is to some document from the corpus or even some collection. This is often employed as a weighting factor in hunts of user-friendliness and text mining data retrieval. The TF/IDF value rises to the number of times each word appears in the document and the number of documents in the corpus that contain the word; this may help to rectify the fact that states that there are few words that appear more frequently in general [12].

TF/IDF is a common schemes today; TF/IDF is utilized by 83% of text-based recommender systems in libraries. Employing common words in TF/IDF, e.g., posts get a significant weight even when they provide no exact information about common words. In TF/IDF, the more familiar a word in the corpus, the less the weight it receives. Thus, weights are received by common words like posts. However, words, that are assumed to carry additional information receive more weight [13]. Beautiful Soup is a library that is used in Python for extracting data from XML and HTML documents. It functions together with the favorite parser to provide ways of navigating, searching and modifying the parse tree [14].

The RE-module offers sophisticated approaches to produce and utilize rational expressions [15]. A regular expression is a sort of formulation that specifies patterns in strings. The title “regular expression” stems from the prior mathematical treatment of routine collections. We are still stuck with the term. Describing regular expressions can provide us with a straightforward way to define a pair of related strings by describing. Accordingly, we compose a pattern to summarize a set of strings. This pattern string can be compiled to effectively determines whether and where a given string matches the design [16,17].

2. Literature Review

The authors [18] mechanically extracted relevant features from reviews of programs (e.g., information about bugs, plugins and prerequisites) and analyzed the sentiment related to each. In this study, three major building blocks: (i) topic simulating, (ii) belief investigation and (iii) summarization port are discussed. The subject modeling block aims to find topics that w semantic from textual remarks, extracting the attributes based on the words of every issue. The opinion analysis block detects the sentiment associated with each discovered feature.

The summarization interface provides programmers [19] with an intuitive visualization of these features (i.e., topics)—along with their associated sentiment—providing more valuable information than a ‘star rating’. Our analysis demonstrates that the topic modeling block may organize the information supplied by users into subcategories that facilitate the comprehension of features that may be positive, negative—along with also a neutral impact on the overall evaluation of the program. For seeing user satisfaction, the authors could see that—in spite of the star rating being an extreme measure of investigation—the sentiment analysis method was more precise in capturing the opinion transmitted from the consumer using a remark.

The authors discussed [20] the sentiment analysis of application reviews. In this study, the result shows the sentiment analysis of app review approaches that are helpful for the developers. With these

approaches, the developer can accumulate, filter and examine user reviews. In this research, the use of simple language techniques to recognize fine-grained application features in the review studies. In this extract, the user gives an opinion about the known features by providing a typical score to all reviews.

The authors discussed [21] the result of the usage of latent semantic analysis, according to the research experiment on two types of English text. The size of the text is equal to approx. 500 KB. The purpose of using the latent semantic analysis (LSA) is to incarceration all the mutual dependencies among words and their context and provide the context of equitable sizes. These methods are equally essential for attaining superior outcomes. The resulting algorithm proves the control of the set test and translates the result as a desired quality.

According to research [22], text-mining techniques were employed to classify and summarize user reviews. However, due to the unstructured and diverse character of basic online information that is user-generated, text-based inspection mining techniques create complicated models that are prone to overfitting. Within this study, the authors suggested approach, based on frame semantics, for inspection mining.

The authors of [23] propose semantic frames that help generalize in your raw text (individual words) to abstract scenarios (contexts). This representation of text is expected to boost the predictive abilities of inspection mining methods and lower the chances of overfitting.

The authors of [24]—in perfecting user testimonials that are educational into various categories of software—ask about maintenance. First, the authors investigated the operation of frames. Second, the authors proposed and evaluated the performance of multiple summarization algorithms in representative and concise summaries of reviews that were informative. Three datasets of application shop testimonials were used to conduct an experimental investigation.

In another research [25], results have shown that semantic frames may enable an economic review classification process that was quick and exact. Nevertheless, in reviewing summarization jobs, our deductions assert that summarization creates comprehensive summaries than summarization. In closing, authors have introduced MARC 2.0, a review classification and summarization package that implements the algorithms investigated in the analysis.

In another research [26], nowadays, the use of apps has increased with the use of advanced mobile technology. Users prefer to use mobile phones for mobile applications over any other devices. Users already downloaded different mobile applications in their mobile phones and they use these applications and left reviews about it.

The authors of [27] research in the mobile application market, fallacious ranking points, may lead to pushing up mobile apps in the popularity list. Indeed, it turns more periodic for application developers to use fake mechanism. The study has, at this moment, proposed a semantic analysis of application review for fraud detection in mobile apps. First, authors have submitted to detect the misrepresentation by excavating the active periods correctly, also called as leading sessions, of the mobile apps [27].

The authors of [28] have an intention to inspect two types of evidence: ranking-based review and -based and use natural language processing (NLP) to get action words. Next, authors agreed to convert reviews to ratings and finally perform pattern analysis on the session with application data gathered from the application store. Hence, the study proposed an approach to validate its effectiveness and show the scalability of the detection algorithm.

The author of [29]—another research—find user reviews to be a part of available application markets like the Google Play Store. The question arises: How do writers make sense from these and summarize millions of consumer reviews? Unfortunately, beyond straightforward summaries like histograms of consumer evaluations, few analytic tools exist that can provide insights into user testimonials.

According to the research [30], this application may (a) find inconsistencies in reviews (b) identify reasons why consumers prefer or dislike a given program and also provide an interactive, zoomable view of evolving user reviews; and (c) offer valuable insights into the entire application market,

differentiating important user issues and preferences of different kinds of programs. The goods on “amazon.com” the mobile application, are constantly evolving, with newer versions immediately superseding the older ones. The App store utilizes an evaluation program, which aggregates each test delegated into a store rating.

Authors describe [31] the researchers resisted the store ratings of more than apps daily, whether the store rating captures the erratic user-satisfaction levels regarding application versions to examine. However, many app version evaluations increased or dropped; their store evaluation was resilient to changes once they had gathered a quantity of raters. The result of ratings is not energetic enough to catch varying user satisfaction levels. This durability is an issue that could discourage programmers from improving program quality.

In this research, authors [32] propose a self-learning base architecture that is used in the analysis process. According to this research, this architecture is best for an analysis of a huge amount of data sources with minimal interference of the user.

In this study, the authors [33] examined the problem of sparse principal component analysis (PCA). The PCA is a tool that is commonly used for the analysis of data as well as for visualization. This study presents an algorithm for a single-factor sparse PCA problem and this algorithm performance is slightly better than the other methods. They use a different type of dataset, for example, news data, voting data, etc. to get their desire results. According to this study, they use convex relaxation methods for good results and for good solution quality greedy algorithm found better in this research.

The authors [34] recommend a framework for discrete records and document modeling of the topical structure. According to the method, this model allocates a word in a text document to a specific topic. There are many-to-many relationship methodologies that are used between the topic and the words, as well as among the topic and documents. LDA makes this model easy, and it can easily use with the complex architecture. This model is not only feasible in the document cluster for the topic, but also reasonable in various dimensions. This model was found to be efficient at improving several traditional models.

In another research, reference [35] propose a method probabilistic latent semantic analysis for unsupervised learning and its base on a latent statistical model. This method is very accurate than normal LSA because of its base on statistical approaches. This research found for fitting procedure tempered expectation maximization is the dominant method. Result prove the study get their desired result and PLSA is an excellent method use with a various application that is based on information extracting and text learning.

The authors of [36] propose a new visualization engine called automatic exploration of textual collections for healthcare (ACE-HEALTH), which is used to analyze medical data using latent Dirichlet allocation (LDA). This engine combines various visualization techniques to provide controllable revealing consoles. The result shows the effectiveness of this engine with its compactness.

The result proposes an encouraging method that is used in the current scenario. The current methods of machine learning are feasible for this dataset and prove better results. Experiments also show the use of these combined features can improve the model of machine learning and improve performance.

3. Materials and Methods

This section consists of the explanation of the dataset used for data collection, its visualization and the proposed methodology used in our research on the chosen dataset.

3.1. Data collection Description

In this research, we scraped thousands of user reviews and ratings of different applications based on different categories, as shown in Figure 1. We selected 14 categories from the Google Play Store, and scraped different applications from each class, as shown in Table 1. These application categories were: action, arcade, card, communication, finance, health and fitness, photography, shopping, sports,

video player editor, weather, casual medical and racing. We scraped thousands of reviews and ratings of application and converted these data into a “.CSV” file format. After this, we are applied a preprocess to remove special characters, remove a single character, remove a single character from the start, subtract multiple spaces with single spaces, remove prefixes, convert data into lowercase. We used stop words and a stemming technique on data in the “.CSV” file. We then evaluated the results by using different machine-learning algorithms to find the best algorithm for classification. We downloaded 148 apps that appeared in 14 categories from the Google Play Store, fetched several reviews and entered the required pages according to the reviews. We collected a total of 506,259 reviews from the Google Play Store website, as shown in Figure 1. To fetch the data, in the first step we used a requests library. We used Python’s Scikit-Learn library (Pedregosa, Varoquaux, Gramfort, Michel, Thirion, Grisel & Vanderplas, 2011) for machine learning because this library provides machine-learning algorithms like classification, regression, clustering, model validation, etc. [32]. The requests library allows the user to send HTTP/1.1 requests using Python to add content like headers. This library enabled users to process response data in Python. Then we used the Re-library for text processing. A regular expression is a unique sequence of characters that help the user match or find other strings or sets of strings, using a specific syntax held in a pattern. After using the Re-library, we used the BeautifulSoup library to extract data from the HTML and XML files. The measurement of scraped results from different categories is shown in Table 1.

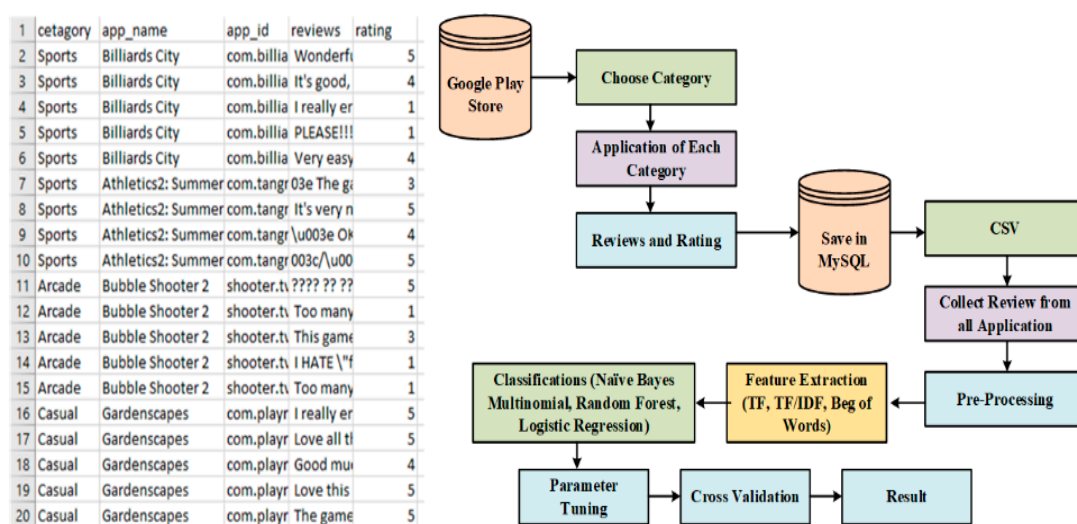


Figure 1. Methodology diagram of data collection and sample shot of dataset.

Table 1. Detail measurements of scraped datasets.

Action		Card		Arcade	
App Name	Reviews	App Name	Reviews	App Name	Reviews
Bush Rush	4001	29 Card Game	4001	Angry Bird Rio	4481
Gun Shot Fire War	3001	Blackjack 21	1601	Bubble Shooter 2	4001
Metal Soldiers	4001	Blackjack	4481	Jewels Legend	4001
N.O.V.A Legacy	4364	Callbreak Multiplayer	4001	Lep World 2	3001
Real Gangster Crime	4001	Card Game 29	3066	Snow Bros	3001
Shadow Fight 2	4481	Card Words Kingdom	4472	Sonic Dash	4481
Sniper 3D Gun Shooter	4481	Gin Rummy	3616	Space Shooter	4401
Talking Tom Gold Run	4001	Spider Solitaire	2801	Subway Princess Runner	3001

Table 1. Cont.

Action		Card		Arcade	
App Name	Reviews	App Name	Reviews	App Name	Reviews
Temple Run 2	3001	Teen Patti Gold	4481	Subway Surfers	4481
Warship Battle	4001	World Series of poker	4001	Super Jabber Jump 3	2912
Zombie Frontier 3	4001				
Zombie Hunter King	3782				
Communication		Finance		Health and Fitness	
App Name	Reviews	App Name	Reviews	App Name	Reviews
Dolphin Browser	3001	bKash	4481	Home Workout—No Equipment	4481
Firefox Browser	3001	CAIXA	1220	Home Workout for Men	1163
Google Duo	3001	CAPTETEB	697	Lose Belly Fat In 30 Days	4481
Hangout Dialer	3001	FNB Banking App	2201	Lose It!—Calorie Counter	4001
KakaoTalk	3001	Garanti Mobile Banking	859	Lose Weight Fat In 30 Days	4481
LINE	3001	Monobank	605	Nike+Run Club	1521
Messenger Talk	3001	MSN Money-Stock Quotes & News	3001	Seven—7 Minutes Workout	4425
Opera Mini Browser	3001	Nubank	1613	Six Pack In 30 Days	3801
UC Browser Mini	3001	PhonePe-UPI Payment	4001	Water Drink Reminder	4481
WeChat	3001	QIWI Wallet	1601	YAZIO Calorie Counter	1590
		Yahoo Finance	3001		
		YapiKredi Mobile	1952		
		Stock	3001		
Photography		Shopping		Sports	
App Name	Reviews	App Name	Reviews	App Name	Reviews
B612—Beauty & Filter Camera	4001	AliExpress	4481	Billiards City	4481
BeautyCam	4001	Amazon for Tablets	4481	Real Cricket 18	3001
BeautyPlus	4001	Bikroy	4481	Real Football	3001
Candy Camera	4481	Club Factory	4001	Score! Hero	3001
Sniper 3D Gun Shooter	4481	Gin Rummy	3616	Space Shooter	4401
Google Photos	4481	Digikala	4001	Table Tennis 3D	3001
HD Camera	4001	Divar	4001	Tennis	3001
Motorola Camera	4001	Flipkart Online Shopping App	4481	Volleyball Champions 3D	3001
Music Video Maker	4001	Lazada	4481	World of Cricket	4481
Sweet Selfie	4481	Myntra Online Shopping App	4481	Pool Billiards Pro	4001
Sweet Snap	4001	Shop clues	4481	Snooker Star	2801
Video Player Editor		Weather		Casual	
App Name	Reviews	App Name	Reviews	App Name	Reviews
KineMaster	1441	NOAA Weather Radar & Alerts	3601	Angry Bird POP	4481

Table 1. Cont.

Video Player Editor		Weather		Casual	
App Name	Reviews	App Name	Reviews	App Name	Reviews
Media Player	2713	The Weather Channel	4001	BLUK	3281
MX Player	3001	Transparent Weather & Clock	1441	Boards King	4481
Power Director Video Editor App	1641	Weather & Clock Weight for Android	4481	Bubble Shooter	4481
Video Player All Format	1041	Weather & Radar—Free	3601	Candy Crush Saga	4481
Video Player KM	3001	Weather Forecast	1681	Farm Heroes Super Saga	4481
Video Show	1321	Weather Live Free	1721	Hay Day	4481
VivaVideo	4190	Weather XL PRO	1401	Minion Rush	4481
You Cut App	1241	Yahoo Weather	4361	My Talking Tom	4481
YouTube	1201	Yandex Weather	1045	Pou	4481
				Shopping Mall Girl	4481
				Gardens capes	4481
Medical		Racing			
App Name	Reviews	App Name	Reviews		
Anatomy Learning	2401	Asphalt Nitro	4481		
Diseases & Dictionary	3201	Beach Buggy Racing	4481		
Disorder & Diseases Dictionary	2401	Bike Mayhem Free	4481		
Anatomy Learning	2401	Asphalt Nitro	4481		
Diseases & Dictionary	3201	Beach Buggy Racing	4481		
Disorder & Diseases Dictionary	2401	Bike Mayhem Free	4481		
Drugs.com	2401	Bike Stunt Master	2745		
Epocrates	1001	Dr. Driving 2	4481		
Medical Image	1423	Extreme Car Driving	4481		
Medical Terminology	1448	Hill Climb Racing 2	3801		
Pharmapedia Pakistan	4134	Racing Fever	4481		
Prognosis	2401	Racing in Car 2	4481		
WikiMed	3201	Trial Xtreme 4	4481		

3.2. Methodology

In this methodology for classification, we started by scraping application reviews. On Google Play Store using the AppID request for the scrape, we scraped the reviews of a specific application for several pages with reviews and ratings of the applications. We have scraped this dataset to classify user reviews that were positive, negative or neutral review. After scraping the bulk raw reviews, the next step was preprocessing of those reviews. In preprocessing different levels, we normalized our reviews after preprocessing. These steps involved removing a particular character, deleting a single character, removing a single character from the start, subtracting multiple spaces with individual spaces and removing prefixes, then converting data into lowercase. At the end of these, stop words and stemming was performed. These were the main steps for refining our reviews. After improving reviews, the bag of words approach was presented. In the next level, apply term frequency (TF) on reviews by using

a Python language. After this, we applied term frequency–inverse document frequency (TF/IDF); this is often used in information retrieval and text mining. After applying TF/IDF, feature extraction was performed on each application. By using Python, we used a different algorithm for classification naïve Bayes, random forest and logistic regression and checked the various parameters like accuracy, precision, recall and F1-score and found the statistical information of these parameters. After analyzing and testing from statistical data, we determined the result of which algorithm had the maximum accuracy, precision, recall and F1-score information. We can see which algorithm is best for analyzing reviews for classification in Figure 2.

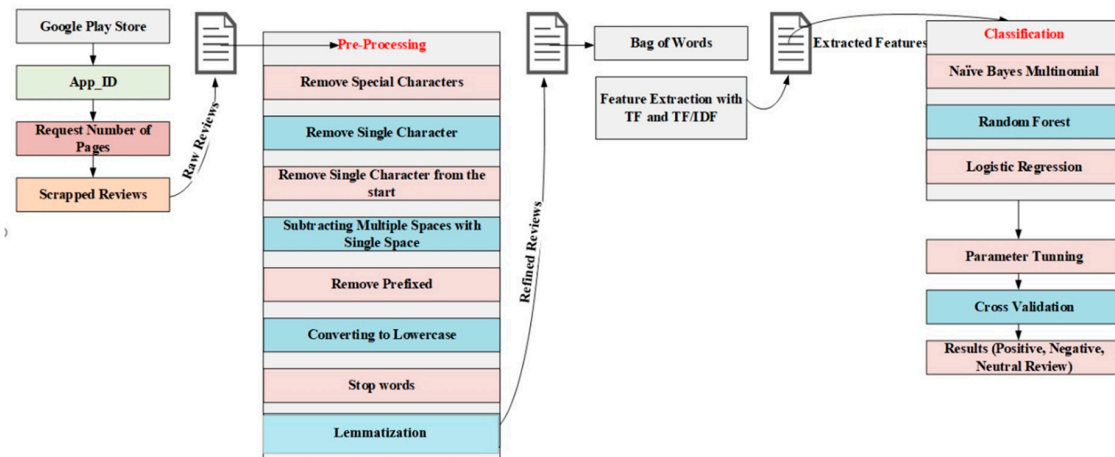


Figure 2. Flow of Google Play Store application reviews classification.

3.3. Supervised Machine Learning Models

In this section, we implement different machine-learning algorithms. For the implementation of machine-learning algorithms, we use the Scikit-learn library and NLTK. These three algorithms mainly use for classification and regression problems. For this purpose, we use the naïve Bayes multinomial, random forest algorithm and logistic regression algorithm.

3.4. Classifier Used for Reviews

This section contains the implementation of different machine-learning algorithms used in this study.

3.4.1. Logistic Regression

Logistic regression is one of the techniques used for classification. Its performance modeling of an event occurring versus event is not occurring. We check the probability of events occur or not occur. In Binary classification, there are two choices, i.e., (0 or 1, yes or not) on the other hand in multiclass Classification, there are more than two categories.

$$\log\left(\frac{P(X)}{1-p(x)}\right) = \beta_0 + \beta_0X_1 + \dots + \beta_pX_p, \tag{1}$$

where $X_1, X_2 \dots X_p$ is independent variables and $B_0, B_1 \dots B_p$ is the coefficients.

$$P = \frac{e^{\beta_0 + \beta_1X_1 + \dots + \beta_pX_p}}{1 + e^{\beta_0 + \beta_1X_1 + \dots + \beta_pX_p}} \tag{2}$$

3.4.2. Random Forest

Random forest is an ensemble model that is well-known for producing accurate prediction results. In the first study [35] about the random forest, which explains some concepts about the ensemble of decision tree known as random forest. Using a single tree classifier, maybe there some problems raised, e.g., outliers, which may affect the overall performance of the classification method. However, because of the uncertainty random forest is one of the types of classifiers that forcefully to outliers and noise [36]. Random forest classifier has two kinds of characters; one is for with rest to data and the second one is for its features. The main features of random forest are bagging and bootstrapping [35].

$$mg(X,Y) = \text{avk } I(\text{hk}(X) = Y) - \max_{j \neq Y} \text{avk } I(\text{hk}(X) = j) \quad (3)$$

3.4.3. Naïve Bayes Multinomial

We now all are aware the naïve Bayes classifier is predicated upon exactly the bag-of-words version. Using the bag-of-words together with all the versions, we assess which word-of this text-document looks in also a negative-words-list or a positive-words-list. In case the term arises within a positive-words-list, this text's rating was upgraded together using all +1 and vice versa. In the end, if the result score is positive, then the text is classified in the category as positive and if it is in negative form, so the text is classified as in negative category [36].

$$P(x|y) = \frac{P(y|x)pP(x)}{P(y)} \quad (4)$$

4. Result and Discussion

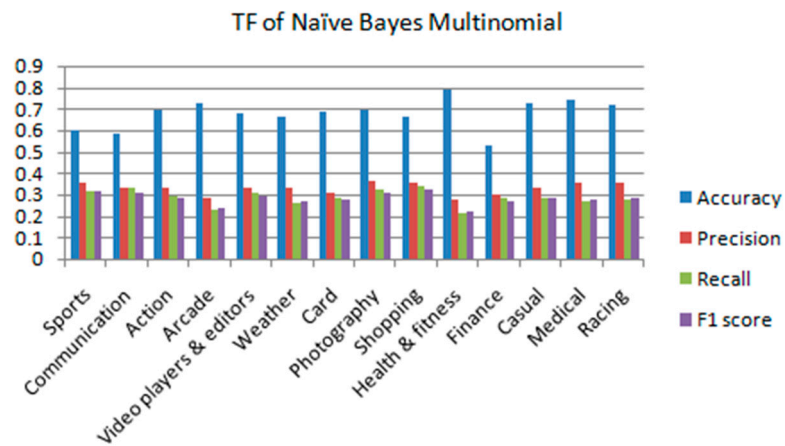
Here, we present statistical information of the different algorithms used in these experiments on the base of the various parameters after preprocessing. We compare the different algorithms and identify the best to be used for classifying and analyzing user reviews.

4.1. Analytical Measurement and Visualization after Preprocessing

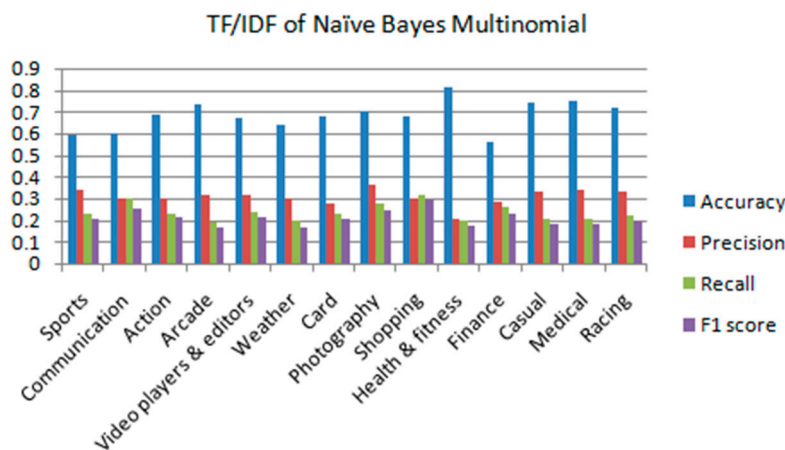
Below are the statistical information of different algorithms on the base of the various parameters after preprocessing. We compare and find the best algorithm that uses for the analysis and classification of reviews.

4.1.1. Naïve Bayes Multinomial

Naïve Bayes was used for classification. It assumes that the occurrence of a specific feature is independent of the occurrence of other features. From a prediction perspective, the performance of this model is considered very fast compared to other models. We scraped 148 apps reviews from 14 categories from Google Play Store. There were 40 reviews on one page, we collected a total of 506,259 reviews from Google Play Store applications. We applied the naïve Bayes algorithm for classification on that dataset of reviews and found different information on different parameters concerning TF and TF/IDF. We calculated the classification accuracy of the model for each application category and reported the precision, recall and F1 scores. Figure 3 illustrates a visualization bar chart for the naïve Bayes algorithm in which series1 indicates the accuracy of the algorithm, series2 indicates the precision, series3 indicates the recall, and series4 indicates the F1 score measurement.



(a)

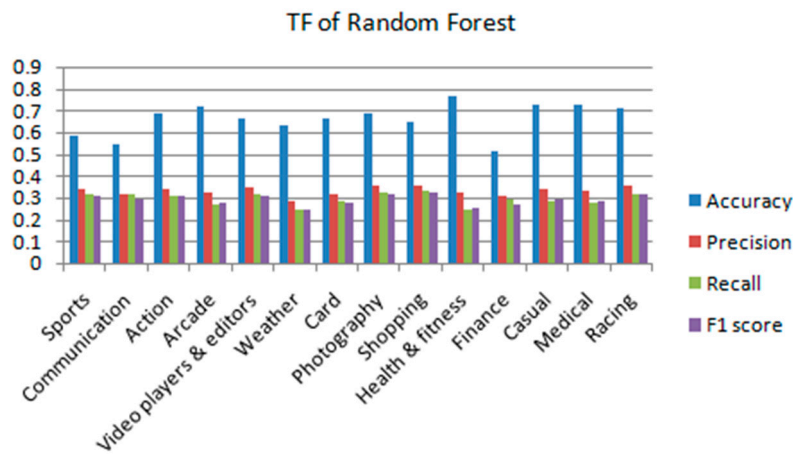


(b)

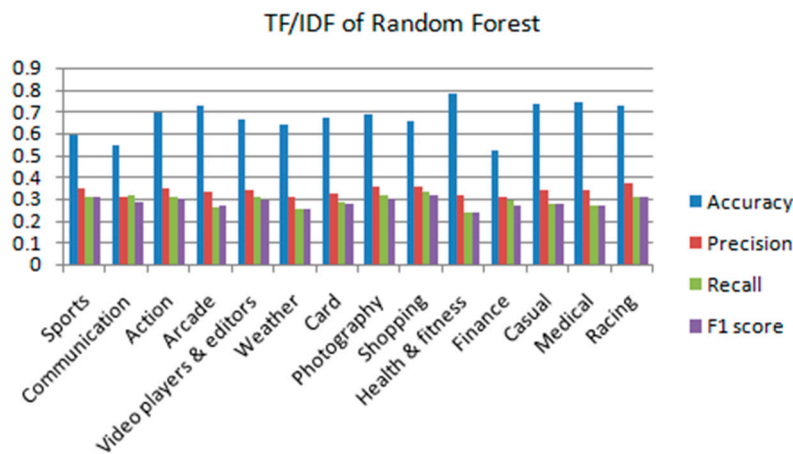
Figure 3. (a) Bar-chart visualization of term frequency (TF) naïve Bayes multinomial algorithm after preprocessing; (b) bar-chart visualization of term frequency/inverse document frequency (TF/IDF) naïve Bayes multinomial algorithm after preprocessing.

4.1.2. Random Forest Algorithm

The Random forests classifier is an ensemble learning method for classification that operates by constructing a multitude of decision trees in which the outcomes are calculated based on the random selection of data. In our experiment, this was done by classifying the reviews and applying different information on different parameters concerning TF and TF/IDF in which we calculated the accuracy of the classification of each application category. In statistical information, we reported precision, recall and F1 scores. Figure 4 illustrates a visualization bar chart for the random forest algorithm in which series1 indicates the accuracy of the random forest algorithm, series2 indicates the precision, series3 indicates the recall, and series4 indicates the F1 score measurement.



(a)



(b)

Figure 4. (a) Bar-chart visualization of the TF random forest algorithm after preprocessing; (b) bar-chart visualization of the TF/IDF random forest algorithm after preprocessing.

4.1.3. Logistic Regression Algorithm

According to the statistics, the logistic regression algorithm can be a reliable statistical version, in which its essential type that runs on the logistic functionality to simulate a binary determining factor; many complex extensions exist. It is an application for binomial Regression. In this experiment, we applied the logistic regression algorithm for classification on the dataset for reviews and found different information on different parameters concerning TF and TF/IDF. We calculated the accuracy of classification of each category application and reported precision, recall and F1 scores. Figure 5 illustrates a visualization bar chart for the logistic regression algorithm in which series1 indicates the accuracy of the logistic regression algorithm, series2 indicates the precision, series3 indicates the recall, and series4 indicates the F1 score measurement.

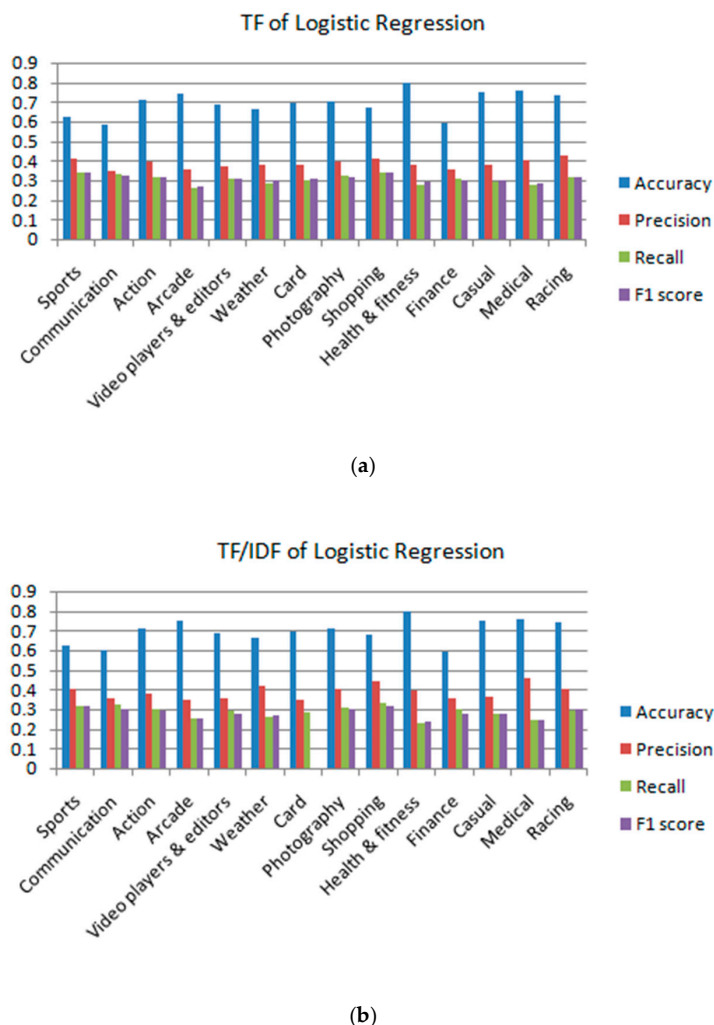


Figure 5. (a) Bar-chart visualization of the TF logistic regression algorithm after preprocessing; (b) bar-chart visualization of the TF/IDF logistic regression algorithm after preprocessing.

4.2. Different Machine-Learning Algorithm Comparison after Preprocessing

The Google Play Store is an online market place that provided free and paid access to users. Google Play Store, users can choose from over a million apps from various predefined categories. In this research, we scraped thousands of user review and application ratings. We evaluated the results by using different machine-learning algorithms like naïve Bayes, random forest and logistic regression algorithm that can check the semantics of reviews about some applications from users to determine if their reviews are good, bad, reasonable, etc. We calculated the term frequency (TF) with different parameters like accuracy, precision, recall and F1 score after the preprocessing of the raw reviews in the concluded results compared the statistical result of these algorithms. We visualized these statistical results in the form of a pie chart, as shown in Figure 6. Figure 7 shows the calculated term frequency (TF) and inverse document frequency (IDF) based result with different parameters after the preprocessing of the raw reviews in the form of pie chart. After comparison, we found that the logistic regression algorithm was the best algorithm for checking the semantic analysis of any Google application user reviews on both TF and TF/IDF bases. As in the sports category in the TF base, we found the logistic regression algorithm had 0.622% accuracy, 0.414% precision, 0.343% recall and 0.343% F1 score, and the statistical information with another category of application, as shown in Table 2. In addition, in TF/IDF base we showed that the logistic regression algorithm had a 0.621% accuracy, 0.404% precision, 0.319% recall and 0.315% F1 score and the statistical information with another category of application, as shown in Table 3.

Table 2. Different machine-learning algorithm comparison of TF-based data after preprocessing.

Application Category	Naïve Bayes Accuracy	Random Forest Accuracy	Logistic Regression Accuracy	Naïve Bayes Precision	Random Forest Precision	Logistic Regression Precision	Naïve Bayes Recall	Random Forest Recall	Logistic Regression Recall	Naïve Bayes F1 Score	Random Forest F1	Logistic Regression F1 Score
Sports	0.602	0.585	0.622	0.359	0.34	0.414	0.316	0.312	0.343	0.315	0.308	0.343
Communication	0.587	0.544	0.585	0.333	0.314	0.349	0.332	0.313	0.329	0.304	0.294	0.32
Action	0.691	0.683	0.707	0.334	0.338	0.395	0.294	0.31	0.312	0.288	0.308	0.313
Arcade	0.725	0.721	0.744	0.283	0.32	0.353	0.231	0.27	0.262	0.235	0.274	0.266
Video players & editors	0.676	0.664	0.684	0.331	0.347	0.37	0.306	0.313	0.306	0.294	0.304	0.304
Weather	0.662	0.632	0.667	0.329	0.285	0.379	0.261	0.243	0.288	0.266	0.248	0.299
Card	0.689	0.665	0.696	0.31	0.312	0.379	0.285	0.285	0.301	0.276	0.279	0.305
Photography	0.696	0.683	0.703	0.367	0.353	0.391	0.327	0.32	0.321	0.31	0.312	0.315
Shopping	0.667	0.648	0.67	0.358	0.354	0.407	0.341	0.333	0.342	0.321	0.324	0.336
Health & fitness	0.788	0.765	0.796	0.273	0.324	0.38	0.212	0.248	0.278	0.218	0.254	0.295
Finance	0.532	0.517	0.592	0.301	0.309	0.352	0.287	0.291	0.311	0.266	0.27	0.303
Casual	0.73	0.728	0.747	0.334	0.341	0.381	0.285	0.284	0.29	0.288	0.292	0.302
Medical	0.745	0.729	0.754	0.359	0.33	0.401	0.272	0.28	0.277	0.279	0.285	0.288
Racing	0.718	0.714	0.737	0.357	0.359	0.428	0.278	0.317	0.312	0.285	0.319	0.318

Table 3. Different machine-learning algorithm comparison of TF/IDF-based data after preprocessing.

Application Category	Naïve Bayes Accuracy	Random Forest Accuracy	Logistic Regression Accuracy	Naïve Bayes Precision	Random Forest Precision	Logistic Regression Precision	Naïve Bayes Recall	Random Forest Recall	Logistic Regression Recall	Naïve Bayes F1 Score	Random Forest F1 Score	Logistic Regression F1 Score
Sports	0.594	0.589	0.621	0.341	0.344	0.404	0.227	0.308	0.319	0.203	0.304	0.315
Communication	0.597	0.545	0.599	0.297	0.307	0.352	0.301	0.312	0.327	0.254	0.288	0.301
Action	0.686	0.691	0.71	0.297	0.347	0.38	0.231	0.306	0.299	0.215	0.302	0.293
Arcade	0.737	0.729	0.747	0.319	0.334	0.351	0.191	0.262	0.25	0.168	0.269	0.252
Video players & editors	0.67	0.664	0.687	0.314	0.34	0.352	0.233	0.304	0.289	0.215	0.295	0.276
Weather	0.642	0.638	0.667	0.301	0.305	0.421	0.194	0.252	0.262	0.168	0.255	0.265
Card	0.68	0.673	0.698	0.28	0.321	0.344	0.227	0.284	0.283	0.209	0.277	0.271
Photography	0.705	0.69	0.71	0.362	0.352	0.405	0.276	0.315	0.311	0.248	0.301	0.297
Shopping	0.678	0.653	0.682	0.299	0.359	0.444	0.316	0.33	0.332	0.289	0.316	0.315
Health & fitness	0.811	0.779	0.801	0.208	0.315	0.391	0.194	0.235	0.23	0.177	0.24	0.235
Finance	0.557	0.52	0.593	0.284	0.31	0.353	0.258	0.293	0.298	0.226	0.27	0.276
Casual	0.745	0.732	0.753	0.334	0.342	0.364	0.205	0.274	0.277	0.182	0.28	0.28
Medical	0.753	0.739	0.759	0.338	0.336	0.459	0.204	0.265	0.244	0.181	0.271	0.245
Racing	0.72	0.724	0.74	0.331	0.37	0.401	0.218	0.306	0.295	0.201	0.311	0.297

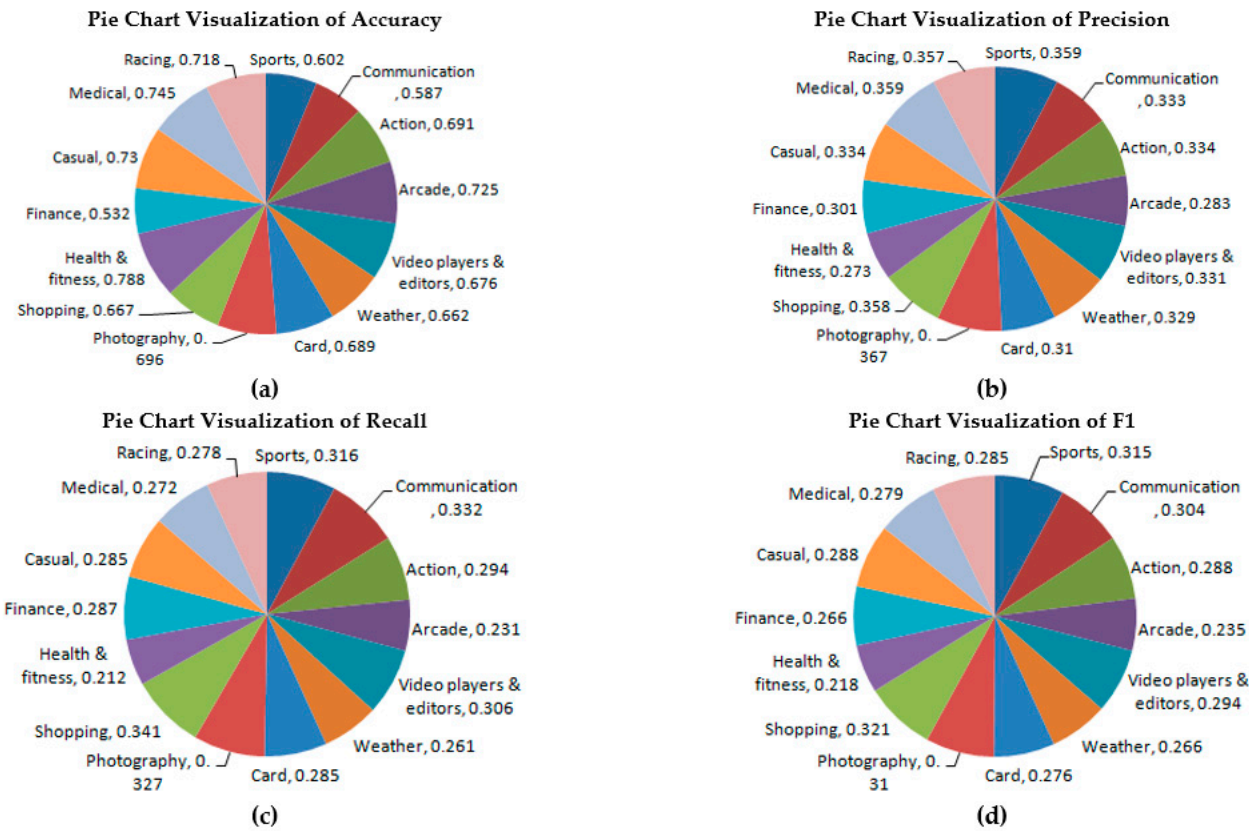


Figure 6. Pie chart visualization of different machine-learning algorithm comparison of TF-based data after preprocessing.

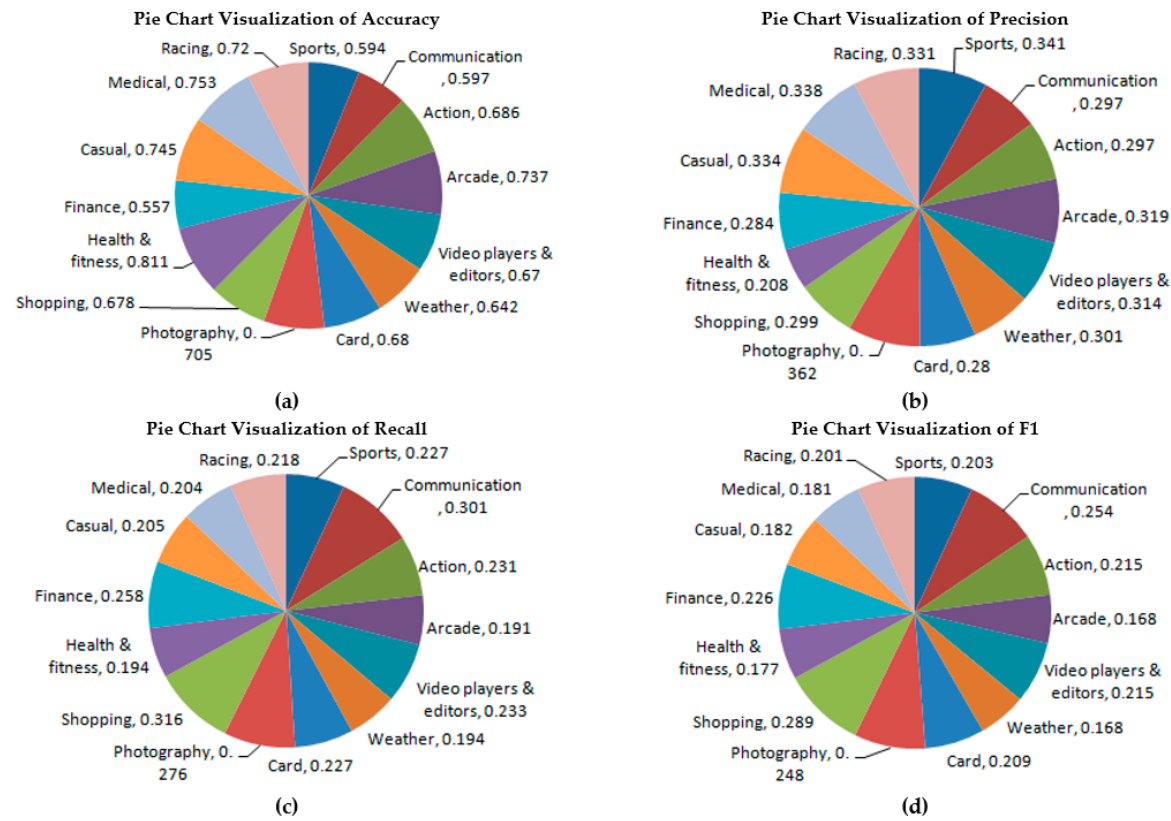


Figure 7. Pie chart visualization of different machine-learning algorithm comparison of TF/IDF-based data after preprocessing.

4.3. Analytical Measurement and Visualization without Preprocessing of Dataset

These are the statistical information of different algorithms on the base of the various parameters after data collection; compare and find the best algorithm that uses for the analysis and classification of reviews.

4.3.1. Naïve Bayes Multinomial

Naïve Bayes is a commonly used classification algorithm. Naïve Bayes assumes that the occurrence of a specific feature is independent of the existence of other features. It is fast to make models and make predictions. We applied the naïve Bayes algorithm for classification on that dataset of reviews and find different information on different parameters concerning TF and TF/IDF. We found the accuracy of classification of each category application, and in the statistical information found precision, recall and F1-scored these all parameters used to measure the accuracy of the dataset. In addition, bar-chart visualization of naïve Bayes algorithm in which series1 shows the accuracy of the naïve Bayes algorithm, series2 shows the precision, series3 shows the recall, and series4 shows the F1 score measurement as shown in Figure 8.

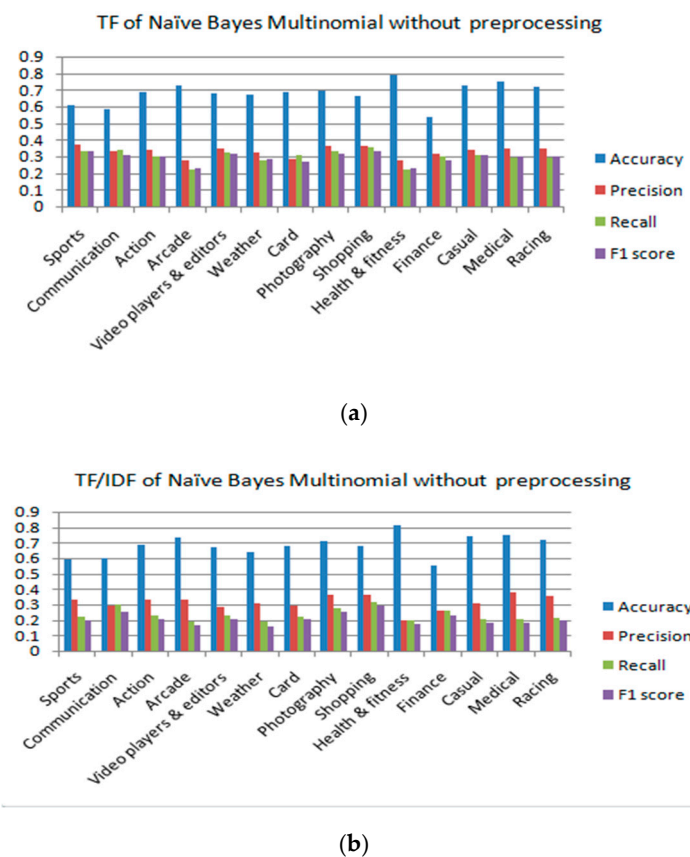
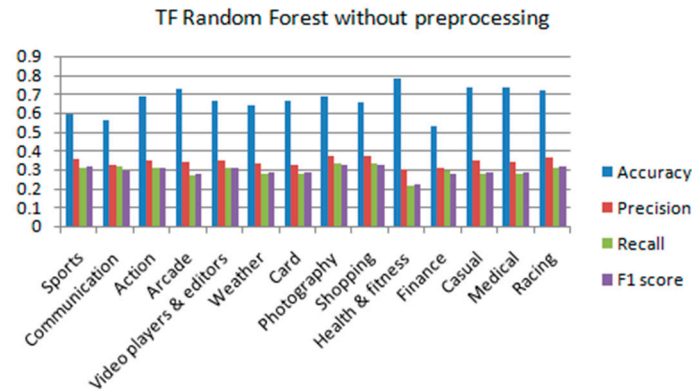


Figure 8. (a) Bar-chart visualization of TF naïve Bayes multinomial algorithm based without preprocessing of data; (b) bar-chart visualization of TF/IDF naïve Bayes multinomial algorithm based without preprocessing of data.

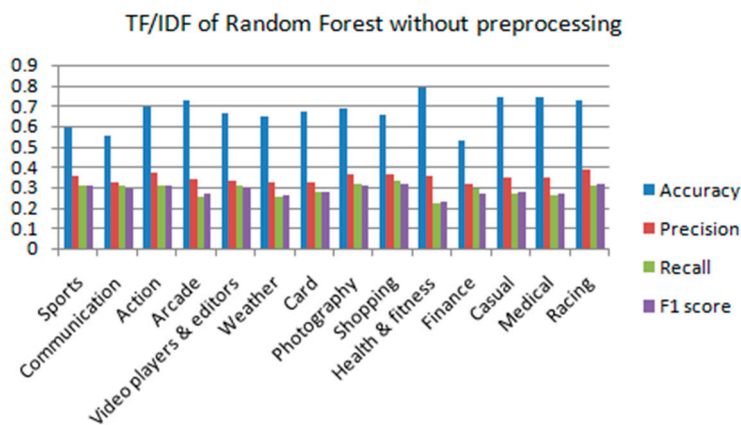
4.3.2. Random Forest Algorithm

The Random forests classifier is the class of all methods that are designed explicitly for decision tree. It develops many decision trees based on a random selection of data and a random selection of variables. We applied the random forest algorithm for classification on that dataset of reviews and find different information on different parameters concerning TF and TF/IDF. Find the accuracy of classification of each category application and in statistical information find precision, recall and

F1 score these all parameters use to measure the accuracy of the dataset. In addition, bar-chart visualization of the random forest algorithm in which series1 shows the accuracy of the random forest algorithm, series2 shows the precision, series3 shows the recall, and series4 shows the F1 score measurement, as shown in Figure 9.



(a)

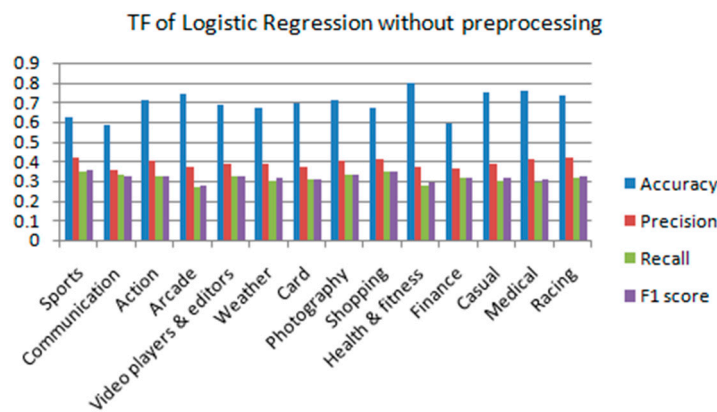


(b)

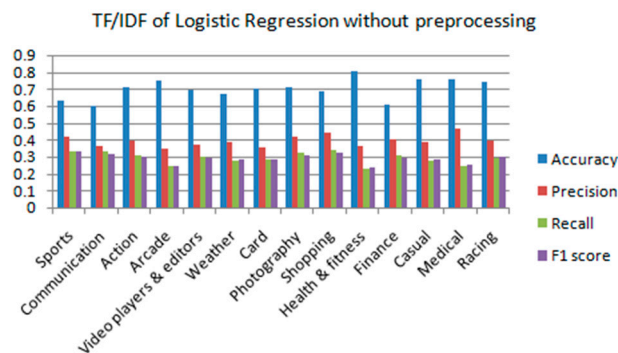
Figure 9. (a) Bar-chart visualization of TF random forest algorithm based without preprocessing of data; (b) bar-chart visualization of TF random forest algorithm based without preprocessing of data.

4.3.3. Logistic Regression Algorithm

In statistics, the logistic product can be a trusted statistical version, which, in its essential type that runs on the logistic functionality to simulate a binary determining factor, many complex extensions exist. Back in Regression investigation, logistic regression will be estimating the parameters of the logistic version; it is an application of both binomial Regressions. We applied the logistic regression algorithm for classification on that dataset of reviews and find different information on different parameters concerning TF and TF/IDF. Find the accuracy of classification of each category application and in statistical information find precision, recall and F1 score these all parameters use to measure the accuracy of the dataset. In addition, bar-chart visualization of the logistic regression algorithm in which series1 shows the accuracy of the logistic regression algorithm, series2 shows the precision, series3 shows the recall, and series4 shows the F1 score measurement as shown in Figure 10.



(a)



(b)

Figure 10. (a) Bar-chart visualization of TF logistic regression algorithm based without preprocessing of data; (b) bar-chart visualization of TF/IDF logistic regression algorithm based without preprocessing of data.

4.4. Different Machine-Learning Algorithm Comparison without Preprocessing of Dataset

Using the Google Play Store, users can choose from over a million apps from various predefined categories. We evaluated the results by using different machine-learning algorithms like naïve Bayes, random forest and logistic regression algorithm that can check the semantics of reviews of some applications from users that their reviews were good, bad, average, etc. We calculated term frequency (TF) of different parameters like accuracy, precision, recall and F1 score after the data collection of the raw reviews in the concluded results compared the statistical result of these algorithms. We visualized these analytical results in the form of a pie chart, as shown in Figure 11. Figure 12 shows the calculated term frequency (TF) and inverse document frequency (IDF) based result with different parameters without preprocessing of the raw reviews in the form of pie chart After comparison, we found that the logistic regression algorithm was the best algorithm to check the semantic analysis of any Google application user reviews on both TF and TF/IDF bases. As in the sports category in the TF base, we show the logistic regression algorithm has 0.623% accuracy, 0.416% precision, 0.35% recall and 0.353% F1 score and the statistical information with another category of application as shown in Table 4. In addition, in TF/IDF base show that the logistic regression algorithm has 0.629% accuracy, 0.416% precision, 0.331% recall and 0.328% F1 score and the statistical information with another category of application as shown in Table 5.

Table 4. Different machine-learning algorithm comparison on TF based without preprocessing of the dataset.

Application Category	Naïve Bayes Accuracy	Random Forest Accuracy	Logistic Regression Accuracy	Naïve Bayes Precision	Random Forest Precision	Logistic Regression Precision	Naïve Bayes Recall	Random Forest Recall	Logistic Regression Recall	Naïve Bayes F1 score	Random Forest F1 Score	Logistic Regression F1 Score
Sports	0.607	0.589	0.623	0.368	0.359	0.416	0.314	0.314	0.35	0.334	0.314	0.353
Communication	0.584	0.559	0.588	0.334	0.321	0.355	0.296	0.296	0.334	0.311	0.296	0.326
Action	0.689	0.686	0.71	0.336	0.35	0.405	0.308	0.308	0.32	0.297	0.308	0.324
Arcade	0.724	0.725	0.744	0.273	0.338	0.369	0.275	0.275	0.271	0.227	0.275	0.278
Video players & editors	0.681	0.665	0.69	0.346	0.351	0.39	0.308	0.308	0.323	0.312	0.308	0.323
Weather	0.669	0.641	0.674	0.327	0.335	0.386	0.282	0.282	0.303	0.281	0.282	0.316
Card	0.689	0.666	0.697	0.282	0.325	0.373	0.281	0.281	0.306	0.272	0.281	0.31
Photography	0.691	0.689	0.707	0.366	0.372	0.403	0.321	0.321	0.332	0.317	0.321	0.328
Shopping	0.663	0.654	0.674	0.364	0.37	0.411	0.325	0.325	0.351	0.333	0.325	0.346
Health & fitness	0.788	0.778	0.794	0.277	0.299	0.369	0.22	0.22	0.28	0.225	0.22	0.295
Finance	0.536	0.532	0.595	0.312	0.311	0.363	0.276	0.276	0.319	0.277	0.276	0.314
Casual	0.727	0.735	0.747	0.338	0.345	0.385	0.284	0.284	0.3	0.306	0.284	0.314
Medical	0.749	0.737	0.757	0.348	0.342	0.41	0.284	0.284	0.295	0.298	0.284	0.31
Racing	0.717	0.719	0.738	0.351	0.361	0.419	0.317	0.317	0.317	0.297	0.317	0.325

Table 5. Different machine-learning algorithm comparison on TF/IDF based without preprocessing of the dataset.

Application Category	Naïve Bayes Accuracy	Random Forest Accuracy	Logistic Regression Accuracy	Naïve Bayes Precision	Random Forest Precision	Logistic Regression Precision	Naïve Bayes Recall	Random Forest Recall	Logistic Regression Recall	Naïve Bayes F1 Score	Random Forest F1 Score	Logistic Regression F1 Score
Sports	0.593	0.595	0.629	0.328	0.352	0.416	0.221	0.307	0.331	0.194	0.309	0.328
Communication	0.597	0.555	0.602	0.292	0.32	0.361	0.302	0.309	0.334	0.255	0.29	0.312
Action	0.683	0.695	0.712	0.329	0.369	0.398	0.227	0.306	0.304	0.208	0.307	0.299
Arcade	0.737	0.729	0.747	0.33	0.336	0.349	0.191	0.256	0.246	0.168	0.265	0.247
Video players & editors	0.669	0.665	0.693	0.281	0.335	0.375	0.229	0.306	0.3	0.208	0.298	0.291
Weather	0.641	0.647	0.674	0.306	0.323	0.384	0.19	0.255	0.275	0.161	0.264	0.282
Card	0.68	0.669	0.699	0.29	0.322	0.359	0.223	0.274	0.288	0.205	0.273	0.281
Photography	0.707	0.69	0.714	0.367	0.363	0.422	0.279	0.318	0.322	0.25	0.308	0.309
Shopping	0.679	0.653	0.686	0.365	0.361	0.444	0.319	0.328	0.339	0.292	0.315	0.324
Health & fitness	0.811	0.788	0.803	0.2	0.354	0.363	0.194	0.22	0.232	0.176	0.225	0.239
Finance	0.554	0.529	0.604	0.261	0.317	0.401	0.257	0.294	0.308	0.226	0.272	0.29
Casual	0.745	0.739	0.755	0.31	0.346	0.385	0.204	0.267	0.28	0.181	0.277	0.285
Medical	0.753	0.743	0.759	0.38	0.351	0.468	0.203	0.259	0.246	0.18	0.268	0.249
Racing	0.718	0.726	0.74	0.359	0.383	0.391	0.214	0.307	0.295	0.195	0.317	0.297

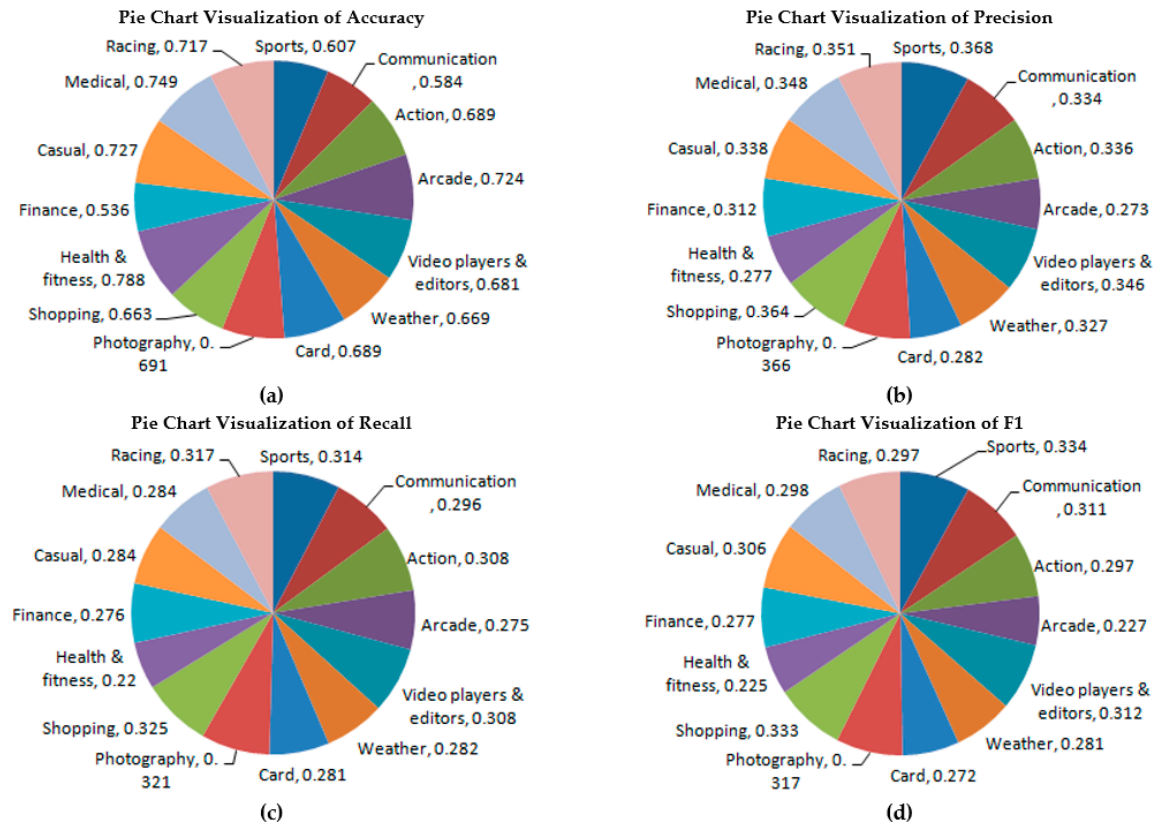


Figure 11. Pie chart visualization of different machine-learning algorithm comparison on TF based without preprocessing of data.

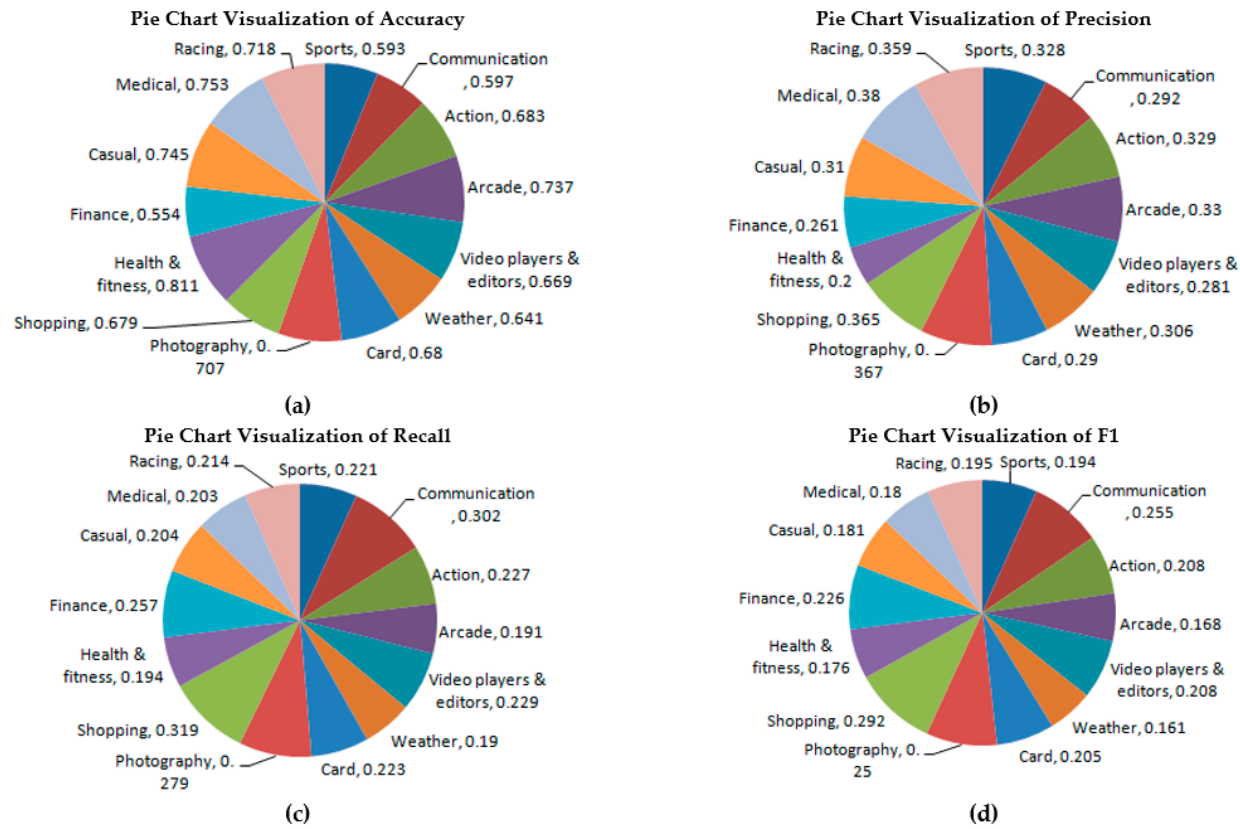


Figure 12. Pie chart visualization of different machine-learning algorithm comparison on TF/IDF based without preprocessing of data.

5. Semantic Analysis of Google Play Store Applications Reviews Using Logistic Regression Algorithm

After checking the different parameters, we analyzed that the logistic regression algorithm was the best algorithm having the highest accuracy. In this section, we performed analysis and classified all reviews in different classes as positive, negative or neutral. Set target value if the value of the comment is positive, it is equal to 1 if the review is negative and it is equal to 0. In addition, we analyzed the neutral class with the confidence rate if the confidence rate is between the 0 and 1 then classified this to neutral class. Different parameters in our dataset like the category of application, application name, Application ID, Reviews and rating are shown in Figure 13. However, for checking the semantics of each review, these parameters were more enough. This is why we selected only reviews of all applications.

1	Cetagory	App_Name	App_ID	Reviews	Rating
2	Sports	Billiards City	com.billiards.city.pool.nation.club	Wonderfull App. Completed all 1020 levels, Can'	5
3	Sports	Billiards City	com.billiards.city.pool.nation.club	It's good, I like the gameplay. Please change up t	4
4	Sports	Billiards City	com.billiards.city.pool.nation.club	I really enjoyed this game until I saw one of the i	1
5	Sports	Billiards City	com.billiards.city.pool.nation.club	PLEASE!!! Get rid of the odd shaped tables and g	1
6	Communication	Hangouts Dialer	com.google.android.apps.hangoutsdialer	st from the notification menu\u003c/\u003e Oth	3
7	Communication	Hangouts Dialer	com.google.android.apps.hangoutsdialer	Wish i found it earlier!!!	5
8	Communication	Hangouts Dialer	com.google.android.apps.hangoutsdialer	sage\u003c/\u003e I love it i dont even have to p	5
9	Communication	Hangouts Dialer	com.google.android.apps.hangoutsdialer	Wanted to make video calling as moto g phone d	1
10	Arcade	Leps World 2	at.ner.lepsWorld2	It is a good time disaster	5
11	Arcade	Leps World 2	at.ner.lepsWorld2	It is so nise i have never seen before	5
12	Arcade	Leps World 2	at.ner.lepsWorld2	I played it totally more than 3 times	5
13	Arcade	Leps World 2	at.ner.lepsWorld2	Is awesome cool game love it some time you lov	5
14	Video Players & Editors	Youtube	com.google.android.youtube	Excellent App	5
15	Video Players & Editors	Youtube	com.google.android.youtube	Very nice	5
16	Video Players & Editors	Youtube	com.google.android.youtube	Very good	4
17	action	WARSHIP BATTLE	com.joycity.warshipbattle	The best at all features this game... Very nice....	5
18	action	WARSHIP BATTLE	com.joycity.warshipbattle	I love this game good work	5
19	action	WARSHIP BATTLE	com.joycity.warshipbattle	I like it	4
20	Weather	NOAA Weather Radar & Alerts	com.apalon.weatherradar.free	Just better than the rest, period.	5
21	Weather	NOAA Weather Radar & Alerts	com.apalon.weatherradar.free	Great tool for the road or around town.	5
22	photography	Sweet Selfie	com.cam001.selfie	It's a very good app	5

Figure 13. Sample screenshot of the original dataset that was scraped.

5.1. Data Preparation and Cleaning of Reviews Steps

HTML Decoding

To convert HTML encoding into text and in the start or ending up in the text field as '&', '"' & ''.'

Data Preparation 2: '#' Mention

"#" carries import information that must deal with is necessary.

URL Links

Remove all URLs that appear in reviews remove them.

UTF-8 BOM (Byte Order Mark)

For patterns of characters like "\xef\xbf\xbd," these are UTF-8 BOM. It is a sequence of bytes (EF, BB, BF) that allows the reader to identify a file as being encoded in UTF-8.

Hashtag/Numbers

Hashtag text can provide useful information about the comment. It may be a bit tough to get rid of all the text together with the "#" or with a number or with any other unique character needs to deal.

Negation Handling

~is the factor that is not suitable in the review to remove them.

Tokenizing and Joining

Parse the whole comment into small pieces/segments and then merge again. After applying the above rules on cleaning, the reviews cleaned the form of reviews, as shown in Figure 14.

1	Reviews
2	wonderfull app completed all levels can not wait for more levels level has bug but you can get around it to complete
3	it good like the gameplay please change up the music as it gets repetitive after the nd level and you can hear the t
4	really enjoyed this game until saw one of the adverts throughout the ngame with man and woman spooning it is ha
5	please get rid of the odd shaped tables and go back to the classic table please fix level where the balls and cue st
6	very easy game to play and has actually given me pointers on how to play nreal game of pool like where to hit the
7	not going to lie just started playing this game probably about hours ago and am thoroughly addicted it pretty awesi
8	level the lower left cushion let ball disappear into nothingness nafter striking the ball it can be seen at different edge
9	people balls do not stop at the same time it wont let the ball go in the pocket that has hand in it the table control tl
10	the gameplay is fun graphics are good and there are lot of levels however the number of ads are insane is there pai
11	great game until you get above level than it starts shooting balls off screen where can not see them or play anymoi
12	just wanted to thank the supervisors creator those in charge whatever powers that be that after days of my compla
13	as pool player this is good app find it helps me with using different english techniques control of the ball as well as
14	just wanted basic billiards game there was this redundant tutorial level up system that was totally unnecessary and
15	installed this game and could not even play it because the game would not load and would shut down waste of tim
16	really enjoyed the older version alot more than now some of these odd ball ntables are more aggravating than enjoy
17	really like the game up to certain point when the normal billiard table is changed to crooked one now its hard to pls
18	pretty decent game but has way too many ads now made me watch second video every time wanted to retry table
19	level and til now no problem but then on this levels the balls keep disappearing under the table and from there there
20	first had trouble getting pass level learned it amp now on level great qame finish there are levels total will go back a

Figure 14. Sample screenshot of the cleaning dataset after preprocessing.

5.2. Find Null Entries from Reviews

It seems there were about 700–800 null entries in the reviews column of the dataset. This may have happened during the cleaning process to remove the null entries with using commands, as shown below.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 400000 entries, 0 to 399999
Data columns (total 2 columns):
text 399208 non-null object
target 400000 non-null int64
dtypes: int64(1), object (1)
memory usage: 9.2 + MB
```

5.3. Negative and Positive Words Dictionary

By using word cloud corpus, I made a negative and positive word dictionary based on the occurrence of words in a sentence to get the idea of what kind of words are frequent in the corpus, as shown in Figure 15.



Figure 15. (a) Positive and (b) negative word dictionary by using the word cloud corpus.

5.4. Semantic Analysis of Google Play Store Applications Reviews Using Logistic Regression Algorithm

In the result, classify all reviews three different classes and can check the confidence rate of each rate that how much that comment is positive, negative and neutral. Set the target value equal to 0 to 1 and check the confidence value in that ratio and check the class of the review using the logistic regression algorithm, as shown in Figure 16.

	Reviews	Tags	Confidence	Target
1				
2	wonderful app completed all levels can not wait for more levels level has bug but you can get around it to complete it by not tou	Positive	0.98	1
3	it good like the gameplay please change up the music as it gets repetitive after the nd level and you can hear the track loop sta	Positive	0.833	1
4	really enjoyed this game until saw one of the adverts throughout the ngame with man and woman spooning it is hardly appropri	Negative	0.997	0
5	please get rid of the odd shaped tables and go back to the classic table please fix level where the balls and cue stick disappear	Negative	0.999	0
6	very easy game to play and has actually given me pointers on how to play nreal game of pool like where to hit the cue ball and i	Positive	0.966	1
7	not going to lie just started playing this game probably about hours ago and am thoroughly addicted it pretty awesome game m	Positive	0.956	1
8	level the lower left cushion let ball disappear into nothingness nafter striking the ball it can be seen at different edges of the scre	Neutral	0.546	0
9	people balls do not stop at the same time it wont let the ball go in the pocket that has hand in it the table control the game que	Negative	0.911	1
10	the gameplay is fun graphics are good and there are lot of levels however the number of ads are insane is there paid version whe	Negative	0.687	1
11	great game until you get above level than it starts shooting balls off screen where can not see them or play anymore really have	Negative	0.998	1
12	just wanted to thank the supervisors creator those in charge whatever powers that be that after days of my complaint am no lon	Positive	0.608	1
13	as pool player this is good app find it helps me with using different english techniques control of the ball as well as learning varic	Positive	0.998	1
14	just wanted basic billiards game there was this redundant tutorial level up system that was totally unnecessary and map thing to	Negative	1	0
15	installed this game and could not even play it because the game would not load and would shut down waste of time and really v	Negative	0.998	0
16	really enjoyed the older version alot more than now some of these odd ball ntables are more aggravating than enjoyable do not l	Negative	0.639	1
17	really like the game up to certain point when the normal billiard table is changed to crooked one now its hard to play the game s	Positive	0.795	1
18	pretty decent game but has way too many ads now made me watch second video every time wanted to retry table no thanks w	Negative	0.977	1
19	level and til now no problem but then on this levels the balls keep disappearing under the table and from there there no way to w	Negative	0.993	1
20	first had trouble getting pass level learned it amp now on level great game finish there are levels total will go back and replay so	Negative	0.522	1

Figure 16. Final sentiment analysis results on Google Play reviews using the logistic regression algorithm.

6. Conclusions and Future Work

On the Google Play Store, users may download as many applications from different categorized groups. In this research, we have considered hundreds of thousands of mobile application user reviews putting into consideration 14 different application categories and downloaded 148 application reviews in which the total number of reviews we worked on was accumulated to 50,6259 reviews out of Google Play Store. Assessed the consequence results using machine-learning algorithms such as naïve Bayes, random forest and logistic regression algorithm, which will evaluate the semantics of applications users’ reviews are equally positive, negative and neutral in which we calculated the term frequency (TF) and inverse document frequency (IDF) with various parameters such as accuracy, precision, recall and F1 score and regarding the statistical impact of those calculations. We did not face any challenges while using TF—especially with common phrases for example, in articles in which we get large weight corpus. In regards to TF/IDF, we noticed that the more a common term from the corpus, the larger the weight it received. Common phrases like posts received weights that were small. However rare words that are assumed to carry details got more weight. In our research, we used the visualization of bar charts to simplify the comparison between the different algorithms and the results achieved out of each one of them. The results reveal that logistic regression could be a perfect algorithm in the term of classification and prediction compared to other algorithms used specially with the dataset used in this experiment and hence, it is worth mentioning that up to our knowledge, this is the first time that such dataset is used for classification In-addition to that, logistic regression got the optimal speed of precision, accuracy, recall and F1 score after preprocessing the dataset. As an example, we were able to achieve good accuracy result for the sport category after preprocessing the dataset and implementing the logistic regression algorithm as in TF 0.622% accuracy, 0.414% precision, 0.343% recall and 0.343% F1 score and in TF/IDF based logistic regression algorithm has 0.621% accuracy, 0.404% precision, 0.319% recall and 0.315% F1 score. In addition, the sports category in TF base after data collection the logistic regression algorithm has 0.623% accuracy, 0.416% precision, 0.35% recall and 0.353% F1 score and in TF/IDF based logistic regression algorithm has 0.629% accuracy, 0.416% precision, 0.331% recall and 0.328% F1 score and the statistical information with another category of applications analyze in the concluded table below that shows the authenticity of this analysis. The section performs analysis and classifies all reviews in different classes positive, negative and neutral. In our research, we set target values of (0 and 1) based on user’s comments where 0 identify negative comments while 1 represents positive comments. However, values between 0 and 1 are set with a confidence rate and considered under the neutral class.

In the future, work, we may consider including more application categories, increasing the number of reviews and comparing the logistic regression algorithm accuracy results with different algorithms.

In addition, we may consider generating clusters and checking the relationship between reviews and ratings of the application that can analyze each application more precisely.

Author Contributions: Conceptualization, A.K.; Supervision, A.A.; methodology, A.K.; validation, A.K., A.A. and S.B.B.; formal analysis, A.K.; investigation, A.K.; resources, A.A., S.B.B.; data curation, A.A.Q.; writing—original draft preparation, A.K.; writing—review and editing, S.B.B., M.A.; visualization, A.A.Q., M.A.; project administration, A.A.; funding acquisition, S.B.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors would like to thanks Qatar National Library (QNL), Qatar, for supporting us in publishing our research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Goldberg, Y. Neural network methods for natural language processing. *Synth. Lect. Hum. Lang. Technol.* **2017**, *10*, 1–309. [[CrossRef](#)]
- Genc-Nayebi, N.; Abran, A. A systematic literature review: Opinion mining studies from mobile app store user reviews. *J. Syst. Softw.* **2017**, *125*, 207–219. [[CrossRef](#)]
- Cambria, E.; Schuller, B.; Xia, Y.; White, B. New avenues in knowledge bases for natural language processing. *Knowl.-Based Syst.* **2016**, *108*, 1–4. [[CrossRef](#)]
- Attik, M.; Missen, M.M.S.; Coustaty, M.; Choi, G.S.; Alotaibi, F.S.; Akhtar, N.; Jhandir, M.Z.; Prasath, V.B.S.; Coustaty, M.; Husnain, M. OpinionML—Opinion markup language for sentiment representation. *Symmetry* **2019**, *11*, 545. [[CrossRef](#)]
- Gao, C.; Zhao, Y.; Wu, R.; Yang, Q.; Shao, J. Semantic trajectory compression via multi-resolution synchronization-based clustering. *Knowl.-Based Syst.* **2019**, *174*, 177–193. [[CrossRef](#)]
- Santo, K.; Richtering, S.; Chalmers, J.; Thiagalingam, A.; Chow, C.K.; Redfern, J.; Bollyky, J.; Iorio, A. Mobile phone apps to improve medication adherence: A systematic stepwise process to identify high-quality apps. *JMIR mHealth uHealth* **2016**, *4*, e132. [[CrossRef](#)]
- Barlas, P.; Lanning, I.; Heavey, C. A survey of open source data science tools. *Int. J. Intell. Comput. Cybern.* **2015**, *8*, 232–261. [[CrossRef](#)]
- Man, Y.; Gao, C.; Lyu, M.R.; Jiang, J. Experience report: Understanding cross-platform app issues from user reviews. In Proceedings of the 2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE), Ottawa, ON, Canada, 23–27 October 2016; pp. 138–149.
- Saura, J.R.; Bennett, D.R. A three-stage method for data text mining: Using UGC in business intelligence analysis. *Symmetry* **2019**, *11*, 519. [[CrossRef](#)]
- Benedetti, F.; Beneventano, D.; Bergamaschi, S.; Simonini, G. Computing inter-document similarity with Context Semantic Analysis. *Inf. Syst.* **2019**, *80*, 136–147. [[CrossRef](#)]
- Sachs, J.S. Recognition memory for syntactic and semantic aspects of connected discourse. *Percept. Psychophys.* **1967**, *2*, 437–442. [[CrossRef](#)]
- Zhang, T.; Ge, S.S. An improved TF-IDF algorithm based on class discriminative strength for text categorization on desensitized data. In Proceedings of the 2019 3rd International Conference on Innovation in Artificial Intelligence—ICIAI, Suzhou, China, 15–18 March 2019; pp. 39–44.
- Zhang, Y.; Ren, W.; Zhu, T.; Faith, E. MoSa: A Modeling and Sentiment Analysis System for Mobile Application Big Data. *Symmetry* **2019**, *11*, 115. [[CrossRef](#)]
- Richardson, L. Beautiful Soup Documentation. Available online: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/> (accessed on 4 May 2015).
- Chapman, C.; Stolee, K.T. Exploring regular expression usage and the context in Python. In Proceedings of the 25th International Symposium on Software Testing and Analysis, Saarbrücken, Germany, 18–20 July 2016; pp. 282–293.
- Chivers, H. Optimising unicode regular expression evaluation with previews. *Softw. Pr. Exp.* **2016**, *47*, 669–688. [[CrossRef](#)]
- Coutinho, M.; Torquato, M.F.; Fernandes, M.A.C. Deep neural network hardware implementation based on stacked sparse autoencoder. *IEEE Access* **2019**, *7*, 40674–40694. [[CrossRef](#)]

18. Di Sorbo, A.; Panichella, S.; Thomas, C.; Shimagaki, J.; Visaggio, C.A.; Canfora, G.; Gall, H.C. What would users change in my app? Summarizing app reviews for recommending software changes. In Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering—FSE 2016; Association for Computing Machinery (ACM), Seattle, WA, USA, 13–19 November 2016; pp. 499–510.
19. Colhon, M.; Vlăduțescu, Ș.; Negrea, X. How objective a neutral word is? A neutrosophic approach for the objectivity degrees of neutral words. *Symmetry* **2017**, *9*, 280. [CrossRef]
20. Al-Subaihini, A.; Finkelstein, A.; Harman, M.; Jia, Y.; Martin, W.; Sarro, F.; Zhang, Y. App store mining and analysis. In Proceedings of the 3rd International Workshop on Mobile Development Lifecycle—MobileDeLi, Pittsburgh, PA, USA, 25–30 October 2015; pp. 1–2.
21. Nakov, P.; Popova, A.; Mateev, P. Weight Functions Impact on LSA Performance. EuroConference RANLP. 2001, pp. 187–193. Available online: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.69.9244&rep=rep1&type=pdf> (accessed on 2 July 2020).
22. Tröger, J.; Linz, N.; König, A.; Robert, P.H.; Alexandersson, J.; Peter, J.; Kray, J. Exploitation vs. exploration—Computational temporal and semantic analysis explains semantic verbal fluency impairment in Alzheimer’s disease. *Neuropsychologia* **2019**, *131*, 53–61. [CrossRef]
23. Kansal, K.; Subramanyam, A.V. Hdrnet: Person re-identification using hybrid sampling in deep reconstruction network. *IEEE Access* **2019**, *7*, 40856–40865. [CrossRef]
24. Jha, N.; Mahmoud, A. Using frame semantics for classifying and summarizing application store reviews. *Empir. Softw. Eng.* **2018**, *23*, 3734–3767. [CrossRef]
25. Puram, N.M.; Singh, K.R. Semantic Analysis of App Review for Fraud Detection Using Fuzzy Logic. 2018. Available online: <https://www.semanticscholar.org/paper/Semantic-Analysis-of-App-Review-for-Fraud-Detection-Puram-Singh/ed73761ad92b9c9914c8c5c780dc1b57ab6f49e8> (accessed on 2 July 2020).
26. Wang, Y.; Liu, H.; Zheng, W.; Xia, Y.; Li, Y.; Chen, P.; Guo, K.; Xie, H. Multi-objective workflow scheduling with deep-q-network-based multi-agent reinforcement learning. *IEEE Access* **2019**, *7*, 39974–39982. [CrossRef]
27. Puram, N.M.; Singh, K. An Implementation to Detect Fraud App Using Fuzzy Logic. *Int. J. Future Revolut. Comput. Sci. Commun. Eng.* **2018**, *4*, 654–662.
28. Yang, W.; Li, J.; Zhang, Y.; Li, Y.; Shu, J.; Gu, D. APKLancet: Tumor payload diagnosis and purification for android applications. In Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security, Kyoto, Japan, 3–6 June 2014; pp. 483–494.
29. Huh, J.-H. Big data analysis for personalized health activities: Machine learning processing for automatic keyword extraction approach. *Symmetry* **2018**, *10*, 93. [CrossRef]
30. Guo, S.; Chen, R.; Li, H. Using knowledge transfer and rough set to predict the severity of android test reports via text mining. *Symmetry* **2017**, *9*, 161. [CrossRef]
31. Cerquitelli, T.; Di Corso, E.; Ventura, F.; Chiusano, S. Data miners’ little helper: Data transformation activity cues for cluster analysis on document collections. In Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics, Amantea, Italy, 19–22 June 2017; pp. 1–6.
32. Zhang, Y.; d’Aspremont, A.; El Ghaoui, L. Sparse PCA: Convex relaxations, algorithms and applications. In *Handbook on Semidefinite, Conic and Polynomial Optimization*; Springer: Boston, MA, USA, 2012; pp. 915–940.
33. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* **2012**, *3*, 993–1022.
34. Hofmann, T. Probabilistic latent semantic analysis. *arXiv* **2013**, arXiv:1301.6705.
35. Di Corso, E.; Proto, S.; Cerquitelli, T.; Chiusano, S. Towards automated visualisation of scientific literature. In *European Conference on Advances in Databases and Information Systems*; Springer: Cham, Switzerland, 2019; pp. 28–36.
36. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]

