www.dergipark.gov.tr
ISSN:2148-3736
El-Cezerî Fen ve Mühendislik Dergisi
Cilt: 9, No:3, 2022 (1038-1050)

El-Cezerî Journal of Science and Engineering
Vol: 9, No: 3, 2022 (1038-1050)
DOI : 10.31202/ecjse.1059822

ECJSE

Research Paper / Makale

Analysing Content Ratings of Google Apps with Ensemble Learning

Ercan ATAGÜN[1a*], Tunahan TİMUÇİN[1b], Serdar BİROĞUL[1c]

[1]Duzce University, Faculty of Engineering, Computer Engineering Department. Duzce/Turkiye
ercanatagun@duzce.edu.tr

Received/Geliş: 18.01.2022 **Accepted/Kabul:** 03.07.2022

Abstract: Google Play was launched under the name of Android Market and made its reputation known all over the world. The mobile application market, which is a package manager developed by Google for Android users, contains applications that appeal to many areas and age ranges. Applications are spread over a wide range of uses. Thus, the amount and size of the data increased, and this situation began to attract the attention of researchers. The excessive increase in the number of applications makes it difficult for parents to follow up on the content. To provide the content rating of applications on Google Play, it is needed to be classified by machine learning methods. In this study, content rating classification was made by analyzing "Category, Rating, Reviews, Size, Installs, Type, Genres, Last Updated, Current Version, Android Version" features of 10757 applications on Google Play, Ensemble Learning methods (Adaboost, Bagging, Random Forest, Stacking), Logistic Regression, Artificial Neural Network, K-Nearest Neighbors algorithms.

Keywords: Ensemble learning, classification, content rating, google apps

Topluluk Öğrenmesi ile Google Uygulamalarının İçerik Derecelendirmelerini Analiz Etme

Öz: Android Market ismiyle piyasaya çıktıktan sonra Google Play ismiyle ününü tüm dünyaya duyuran, Google'ın Android kullanıcıları için geliştirdiği bir paket yöneticisi olan uygulama marketi, içerisinde birçok alana ve yaş aralığına hitap eden uygulamalar bulundurmaktadır. Uygulamalar geniş bir kullanım alanına yayılmıştır. Böylece verinin miktarı ve boyutu artmış, bu durum araştırmacıların dikkatini de çekmeye başlamıştır. Uygulama sayısındaki aşırı artış ebeveynlerin içerikler konusunda takibini zorlaştırmaktadır. Google Play üzerindeki uygulamaların içerik kontrolünün (content rating) sağlanabilmesi için makine öğrenmesi yöntemleri ile sınıflandırılmasına ihtiyaç duyulmaktadır. Bu çalışmada Google Play üzerindeki 10757 uygulamanın Category, Rating, Reviews, Size, Installs, Type, Genres, Last Updated, Current Version, Android Version özellikleri, Ensemble Learning yöntemleri (Adaboost, Bagging, Random Forest, Stacking), K-Nearest Neighbors, Logistic Regression ve Yapay Sinir Ağı algoritmaları ile analiz edilerek content rating sınıflandırılması yapılmıştır.

Anahtar Kelimeler: Topluluk öğrenme, sınıflandırma, içerik derecelendirme, google uygulamaları

1. Introduction

The application marketplace, Google Play, which is a package manager developed by Google for Android users, contains applications that appeal to many areas and age ranges. Applications in Google Play are increasing day by day. The increase in the number of mobile devices and the ease of access to mobile devices has attracted the attention of individuals under the age of 18. The easy access of these individuals, who can be called children, to mobile applications worries parents. These individuals have the opportunity to access all applications that are suitable or unsuitable for them.

How to cite this article
Atagün E., Timuçin T., Biroğul S., "Analysing Content Ratings of Google Apps with Ensemble Learning", El-Cezerî Journal of Science and Engineering, 2022, 9 (3); 1038-1050.

Bu makaleye atıf yapmak için
Atagün E., Timuçin T., Biroğul S., "Analysing Content Ratings of Google Apps with Ensemble Learning", El-Cezerî Fen ve Mühendislik Dergisi, 2022, 9 (3); 1038-1050.
ORCID: [a]0000-0001-5196-5732; [b]0000-0003-0332-4118; [c]0000-0003-4966-5970

Although there are benefits to being so easy to access applications, there are significant disadvantages such as open content, cyberbullying, and financial losses. Since the increase in the number of applications made parental control more difficult [1, 2], the harm became more pronounced. To solve this problem, it is necessary to classify the applications and ensure that harmful contents are distinguished. In today's technology, this process is done through machine learning algorithms. In this study, the Google Play applications classification problem is solved by using algorithms of Ensemble Learning, which is one of the methods of Machine Learning. A literature review is included in section 2 of the study, the methods and materials used in section 3 are explained, application and experimental results are obtained in section 4, and the study is concluded by giving results in section 5. Parental control becomes more difficult depending on the number of mobile applications.

2. Related Works

While developing an application by software developers, the primary goal is to develop the application in a user-friendly manner and add remarkable features to ensure that it is used by large masses. Although it is not possible to predict the success of an application, it is possible to make highly accurate predictions with various algorithms. In one of these studies, Maredia estimated whether apps on the Google Play Store would be successful with 13 different attributes. Decision Tree, K-Nearest Neighbors (KNN), Naive Bayes, and Logistic Regression (LR) algorithms were used and the highest accuracy was obtained with the Decision Tree algorithm [3]. In another study, Wang et al. studied the characterization of 1.5 million and 2.1 million applications, respectively, removed from Google Play in 2015 and 2017 [4]. With this characterization, it reveals a foresight for both the platform owner and the application developer. Mueez et al., on the other hand, evaluated the applications on Google Play before they were uploaded to the platform and predicted the success of the application by using data such as the number of words in the application name [5]. Kılınç et al. focused on the features that can bring success for application developers. In the study, a developed application was evaluated within the scope of business intelligence and a success estimation was made. Decision Tree, Random Forest, KNN, and Adaboost algorithms were used for regression for rating estimations of applications [6].

When the number of data is so large, the study area expands at a similar rate. E.g; the studies in which the application's scores on the market are analyzed have recently gained momentum.

Sadiq et al. conducted a Deep Learning-based study on the discrepancy between users' comments and ratings on Google Play and Apple Store [7]. Umer et al., by extracting data from 14 application categories on Google Play, worked on estimating the application scores with machine learning techniques and ensuring the consistency of user comments and user votes [8]. Bashir et al. focused on application developers for Google Play and estimated the user rating and the number of installs of applications [9]. Thus, the success of an application will be estimated before it is included in the platform. One of the important analysis studies in the Android Market belongs to Amanullah et al. They showed the effects of price, the number of downloads, and app score on each other among the 10k apps they analyzed [10]. Garg et al., in their studies, studied the issues of applications in Google Play not meeting the claimed content or accessing their data without permission from the user. They combined their analysis with users' responses to such situations [11]. Magar et al., using various classification models, compared and presented the factors affecting the success of an application based on various parameters [12]. Shaw et al. reverse-engineered the application as it is known that it is impossible to predict its score before it is marketed. They compared 1000 applications with the highest score and 1000 applications with the lowest score out of 10740 applications they received on the Slide Me market and aimed to extract the influential features [13].

Although the number of applications developed in the field of health is not sufficient, it has been revealed as a result of studies that existing applications do not work efficiently.

Kaboha et al. tried to identify applications related to elderly care among 4300 applications they scanned on the play store. Although they identified many applications related to elderly care in the market, they stated that these applications were insufficient in terms of "care" and needed to be improved [14]. Ahmed et al. analyzed the efficiency of 5881 drug tracking applications purchased from the market. They determined that only 4 applications work efficiently on these applications and showed the lack of mobile applications in the field of health [15]. Sambhi et al tried to identify applications promising clinical treatment of acne on a total of 614 market application scans. They stated that 25 of these applications focused on providing acne and its treatment, and showed the average downloads and scores in the market [16]. Savic et al. scanned the applications on Google Play for getting rid of addiction. They also shared an analysis of the dependency types and application fees on 87 applications they identified [17]. Krishnan and Selvam performed regression analysis on 5557 apps matching the “diabetes” tag on the play store. As a result of this study, the factors affecting the application download are shown [18]. Biviji et al. performed regression analysis on a total of 634 applications on maternal and infant health on the android market. In this study, they aimed to examine the relationship between maternal and infant health practices and user satisfaction [19]. Ayyaswami et al., in their study in which they revealed the inadequacy and uselessness of applications in the field of health, examined 206 applications in the market for patient education for Atrial Fibrillation (AF). More than half of these applications had to be excluded from the analysis because it was irrelevant, inaccessible, and non-English language. According to the study, the remaining applications are inadequate, illegible, and scientifically unproven [20]. Chyjek et al., examined the applications on the pregnancy wheel and had to exclude them from the analysis because more than two-thirds of the applications included were insufficient and incorrect. This study once again revealed the inadequacy in the field of health [21].

However, there are also areas where there are applications that work efficiently in the field of health, albeit rare.

Frie et al., in the search made on the android market with the tags "weight loss" and "weight tracking", found 179 applications. As a result of the analysis of these applications, together with their scores and 569 user comments, they revealed that wide-ranging applications work efficiently [22].

There are also researchers working in many different areas such as malware detection using Google Play applications, noisy application detection, multi-label applications, and the most used application types.

One of the important studies in the field of malware detection belongs to Takawale and Thakur [23]. They emphasized that the number of Android devices is about 4 times the number of iOS devices in their studies, and they developed an application called Talos, which detects Android malware with machine learning methods. They stated that Talos has an accuracy value of 93.2%. Garg and Baliyan used 85000 applications from 4 different markets to detect malware on android devices using machine learning methods [24]. Mealings and Beach examined the effects of noise monitoring applications on existing smartphones on 499 applications accepted in Australia. They found that 47 of these applications included behavior modification techniques [25]. Siddiqui et al. developed a study to determine the profiles of orthodontic applications on Google Play and the App Store [26]. McIlroy et al., on the other hand, examined the structure of multi-label reviews in 20 mobile applications on Google Play and Apple App Store. In this way, they worked on the correct labeling of user comments [27]. In his study, Kolakaluri aimed to predict which type of applications are mostly used by analyzing the applications in the market with K-nn and SVM classification algorithms. It achieved a precision of 91.4465 % in classification [28].

By analyzing the applications on Google Play, meaningful data that can be used in real life are produced and improvements are made in many areas with this data. Meacham et al., in their study, analyzed 100 applications by searching “vape” and “vaping” on Google Play. By examining the metadata of the application such as title, purpose, and application images, he revealed the relationship of applications to nicotine and other substances [29]. Mahmood focuses on the effectiveness of the methods that app developers use to achieve higher ratings, such as interesting titles, and challenging icons. Random Forest, Linear Regression, and Support Vector Machine algorithms were used to determine the effectiveness of these methods [30]. Malavolta et al., on the other hand, analyzed the reviews of 11917 hybrid applications on Google Play and working on other platforms and revealed their features [31].

3. Method and Material

The applications in the market are increasing day by day and new working areas are emerging. The increase in the number of applications makes it very difficult to monitor the suitability of applications for age groups. In this study, it reveals the success of 10757 applications purchased from Google Play in classification according to content with Ensemble Learning.

3.1. Ensemble Learning

Ensemble Learning aims to improve the performance of a single predictive model by training multiple models and combining their predictions. Ensemble Learning is often tasked with combining multiple predictors in supervised machine learning problems. A model called Base Learner produces a model by taking labeled data as input and generalizing these examples. It can be a Base Learner or an inductive decision tree, a neural network, or a traditional machine learning algorithm such as a regression model. The primary purpose of Ensemble Learning is to combine multiple models to compensate for the errors of a single model with other models and to increase the overall performance of the community above the performance of a single model. Intuitively, Ensemble can be explained as the bringing together of different views, which is human nature, and the tendency to weigh and combine ideas during complex decision-making [32]. Some models that do a local search may get stuck at the local optimum. Combining several models reduces the probability of obtaining the local minimum. Ensemble Learning has many methods in it.

3.1.1. Adaboost Algorithm

Adaboost algorithm is based on recursive training of basic machine learning algorithms on training sets with different distributions [33]. The basic machine learning algorithms are combined into a single and powerful classifier algorithm [34]. In the Adaboost algorithm, the weight values of the relevant samples are increased iteratively. The adaptive Boosting algorithm, also known as Adaboost, is a widely used Ensemble Learning algorithm developed by Freund and Schapire [35]. Boosting methods are based on the collection of weak classifier algorithms and the determination of the intended output with this sum [36]. In this case, the estimation of more than one classifier algorithm is not combined with a classifier.

3.1.2. Bagging Algorithm

Bagging, another Ensemble Learning method, is a collection of independent models in which each Base Learner is trained on backups of samples taken from the original data set. The Bagging algorithm [37] first generates subsets by randomly replacing the training set. With the help of subsets, classifiers are created for learning. The outputs of the classifier models are made by majority voting and the predicted class value of the sample determined to classify is determined [33].

In the Bagging algorithm, the models are independent of each other and therefore it is an Ensemble Learning algorithm that operates in parallel. The boosting algorithm, on the other hand, is a sequential

Ensemble Learning algorithm since it is designed to correct the previous model by focusing on the misclassification operations of the previous model [38].

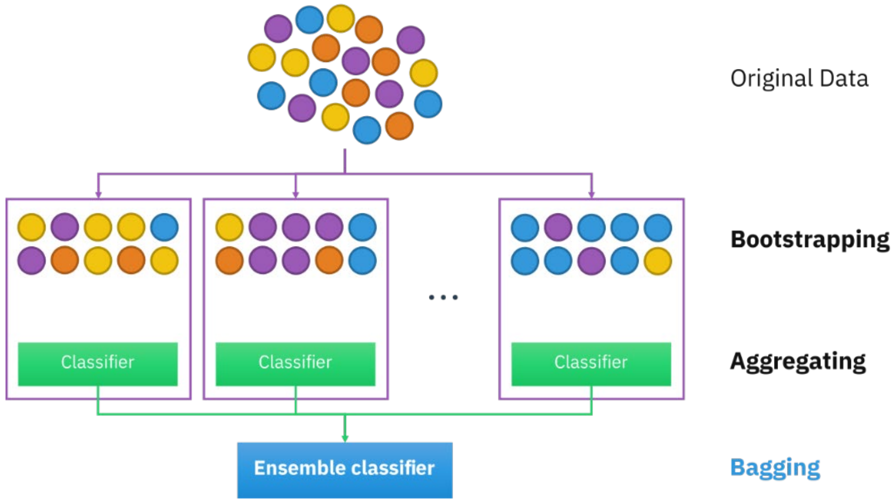

Figure 1. Bagging model

Figure 1 shows the model of the Bagging algorithm.

3.1.3. Random Forest

One of the most popular and widely used Ensemble Learning algorithms is the Random Forest algorithm. Decision trees are trained on different samples and the best partition is tried to be selected.

Random Forest is an Ensemble Learning algorithm that consists of random trees selected from the training dataset independently and with the same distribution [39, 40]. Random Forest[41,42] is one of the machine learning algorithms used in classification and regression problems by creating many decision trees in the learning phase.

One of the most important reasons for choosing the Random Forest algorithm is that it has solved the overfitting problem in the decision tree [36]. Trees are developed with random feature selection by generating data from the original data set to create alternatives to each other [43,44].

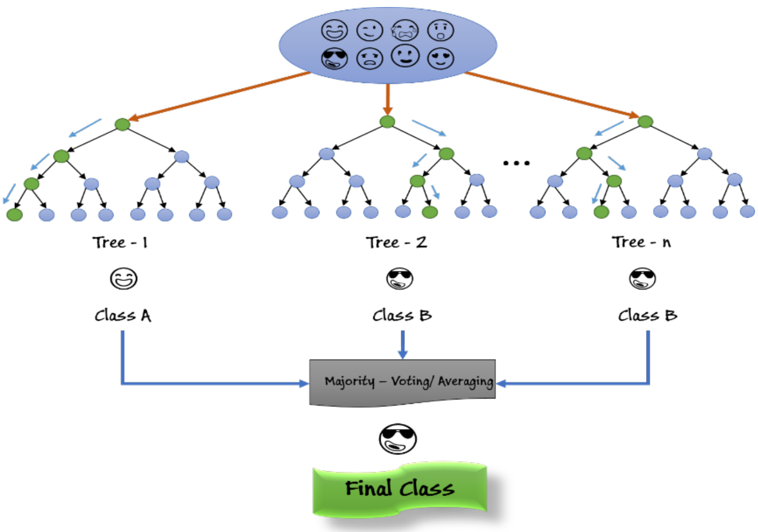

Figure 2. Classic random forest model.

3.1.4. Stacking

The stacking method is used as a meta-learner and establishes a model by determining which model is more reliable [32].

Appropriate classifiers should be included in the model to achieve high performance in Ensemble Learning algorithms. Stacking is one of the algorithms that offer higher performance. In the stacking method, each classifier transmits the estimated value to the meta classifier, and the meta classifier processes the incoming estimated values and produces a final estimate [45].

In this study, Logistic Regression, KNN, and Artificial Neural Networks were included in the Stacking Ensemble Learning model.

The pseudocode of the algorithm is shown [46]:

PSEUDOCODE OF STACKING ALGORITHM

Input: get train samples nj E A
Output: prediction zj j

```
Begin
    Selection M algorithms (S1, S2, S3,…,SM); i = 1;
    while (i<M)
    Training Ei = Si(A) with cross validation
    Zij = Ei(Xj);
End
Generate a data set G with all predictions zij
Train U = S(G) with cross validation;
Zfj = U(G)
```

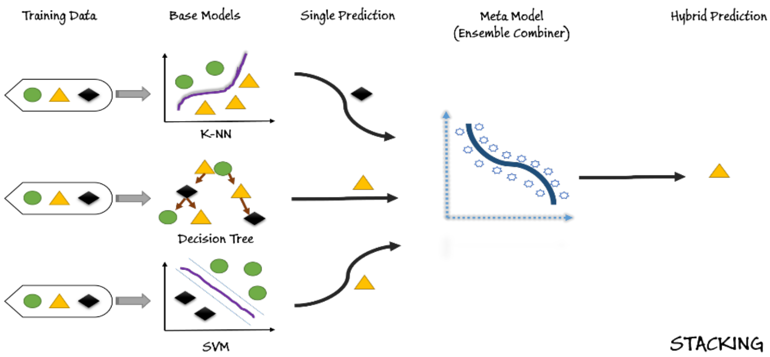

Figure 3. The architecture of the stacking algorithm

Figure 3 shows the architecture of the Stacking Algorithm. The stacking algorithm is based on the combination of different predictors and a meta-model combining these predictions. In this study, Logistic Regression, KNN, and Artificial Neural Network algorithms are included as classifiers in the Stacking algorithm.

3.2. Classic Machine Learning Algorithms

In this study, it is provided to compare the Ensemble Learning model with the most known artificial intelligence algorithms to determine its effectiveness. Results were obtained from the most widely used algorithms such as KNN, Logistic Regression, and Artificial Neural Network.

3.2.1. KNN

This algorithm proposed by Cover and Hart [47] is based on the selection of a selected point according to its group and the number of nearest neighbors [48,49] to this selected point. KNN is within the scope of supervised learning techniques. It uses metrics such as Manhattan, Euclidean, and Cehbysev for distance measurement. While applying the KNN algorithm to the problem, the Euclidean distance metric was used.

$$K(x,y) = \left(\sum_{k=1}^{n} (x_k - y_k)^2\right)^{\frac{1}{2}} \quad (1)$$

3.2.2. Logistic Regression

Logistic Regression makes a value estimation between 0 and 1 by assigning a value as success or failure while estimating an event [50]. If the estimated number of independent variables is more than 2, it is called multivariate regression. This algorithm aims to find the most suitable coefficients of the dependent variables.

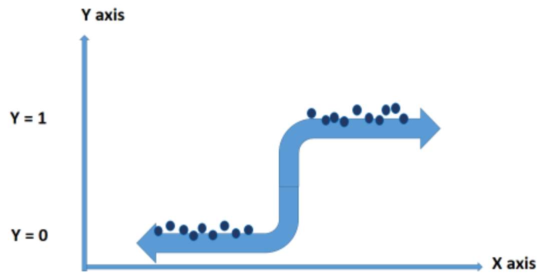

Figure 4. Logistic regression

Figure 4 shows a graphical representation of the Logistic Regression algorithm. When logistic regression is applied to a problem, if the estimated categorical variable is more than 2 values, multinominal logistic regression is used. In this study, multinominal logistic regression was used because the independent variable was more than 2.

3.2.3. Artificial Neural Network

Learning is an ability found in biological creatures and other intelligent systems. Learning in artificial systems is the updating of the internal dynamics of the system in response to external stimuli to fulfill a specific purpose. Artificial Neural Networks update the rules and architecture by incorporating training data into the network by learning from past experiences [51].

In this study, the Artificial Neural Network model was prepared using 200 neurons in the hidden layer, Relu as the activation function, Adam for the Solver parameter, and 10000 for the iteration number.

3.3. Dataset

The dataset used in this study contains information on 10757 applications on Google Play. The "Category" variable is categorical and contains 34 different values. This variable is categorized as ART_AND_DESIGN, GAME, FINANCE. The most common value is FAMILY and 1957 pieces. The "Rating" variable is numeric. Its minimum value is 1 and its maximum value is 5. Its average value is 4192. The "review" variable is numeric. Its minimum value is 0, its maximum value is 78158306. The "size" variable is a string. The "Installs" variable is a string. The variable "Type" is categorical. There are types Free or Paid. The variable "Genres" is categorical. It has 119 different values. This variable is categorized as Medical, Parenting, and Shopping. The "Last Update" variable is categorical. It took 1371 different values. The variable "Current Ver" is categorical. It has 2821 different values. The "Android Ver" variant is categorical. This variable takes 33 different values. Mosaic notation is generally used to show the state table of two or more categorical variables.

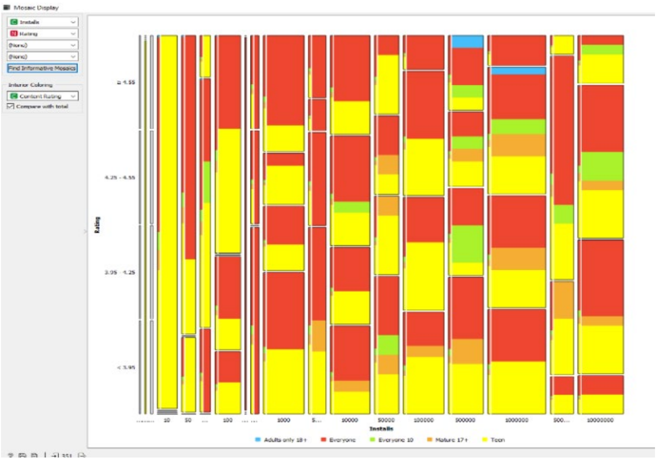

Figure 5. Relationship between content rating and category and rating

Figure 5 shows the relationship between the category and rating attributes of the estimated content rating class. It is observed that the frequency of the Everyone class in the data is higher than the other values. With the mosaic representation, the interaction of 3 different variables is observed on a graphic. The dependent variables named Installs and Rating and the independent variable named Content Rating in the dataset are located on the visual. Mosaic representation can be preferred in the representation of categorical variables.

4. Experimental Results and Discussion

In this section, the classification performance of Google Play applications according to the Content Rating argument is evaluated. The data used in the study were taken from Kaggle [52].

In this study, Content Rating estimation was made using KNN, Artificial Neural Network, Logistic Regression, Adaboost, Stacking, Bagging, and Random Forest algorithms. In the study, 10757 data were divided into 80% training and 20% test data. Accuracy, F1, Precision, and Recall were used for comparison in the study, and also a Receiver Operating Characteristic (ROC) curve was drawn.

Table 1. Algorithms and Parameters

Algorithm	Parameter Name	Value
Logistic Regression	Regularization Type	Ridge(L2)
Logistic Regression	Strength	C=5
Knn	Number of neighbors	5
Knn	Metric	Euclidean
Knn	Weight	Uniform
ANN	Neurons in hidden layers	200
ANN	Activation	ReLu
ANN	Solver	Adam
ANN	Regularization	0.0001
Random Forest	Number of trees	10
Stacking	Learners	Knn, Neural Network, Logistic Regression
Adaboost	Number of estimators	50
Adaboost	Classification	SAMME.R
Bagging	Classifier	REPTree

Table 1 shows the algorithms and their parameters.

Table 2. Classification results.

Model	CA	F1	Precision	Recall
KNN	0.799	0.754	0.745	0.799
Artificial Neural Network	0.836	0.828	0.823	0.836
Logistic Regression	0.803	0.715	0.645	0.803
Stacking	0.853	0.835	0.834	0.853
Random Forest	0.866	0.847	0.852	0.866
Adaboost	0.82295	0.901	0.821	0.998
Bagging	0.8313	0.906	0.854	0.964

Table 2 shows the performance of the algorithm results. It is observed that the highest performance is obtained with the Random Forest algorithm. The Random Forest algorithm is followed by the Stacking algorithm. The Precision value shows how many of the predicted values as "Positive" are actually "Positive". The algorithm with the highest precision value is the Random Forest Algorithm. In terms of precision, it is observed that Ensemble Learning algorithms such as Stacking, Random Forest, and Bagging give higher results. The lowest value for precision was obtained with classical algorithms such as KNN and Logistic Regression. The recall value shows how many of the values that should be estimated as "Positive" are "Positive". The highest recall value was found in algorithms such as Adaboost, Bagging, Stacking, and Random Forest. Classical machine learning algorithms have lower recall values.

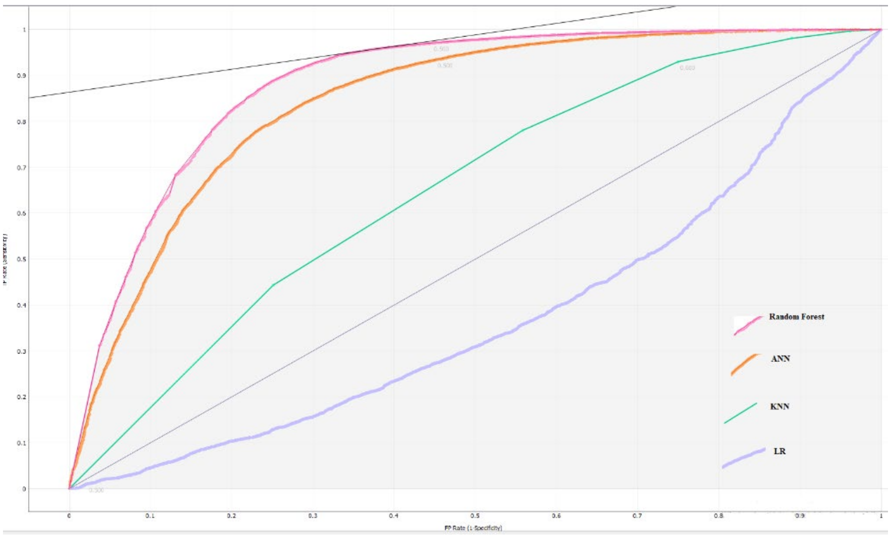

Figure 6. ROC curve.

The F1 metric can be preferred for comparison when the target variable is not evenly distributed in the data set. When the experimental results are examined by the F1 metric, it is observed that Adaboost, Bagging, Random Forest, and Stacking algorithms have higher values. It is observed that the lowest classification performance belongs to classical machine learning algorithms. It is observed that classification performance increases with Ensemble Learning algorithms. It reduces the amount of error in classification by making decisions together with different predictive decision trees in the Random Forest structure. In addition, classification for Google Play data, which has many categorical data types, has been another advantageous situation for Random Forest. Stacking algorithm, on the other hand, achieved higher success than KNN, Artificial Neural Network, and Logistic Regression algorithms that classify alone.

Receiver Operating Characteristic (ROC) analysis is a statistical measurement tool used in performance benchmarking. The ability to predict the content rating class on Google Play applications is expressed by the area under the ROC curve.

Figure 6 shows the ROC curve. Another performance measure used in machine learning problems is the ROC Curve. The ROC curve is used as a benchmark in addition to the precision and recall criteria. As the area under the ROC curve gets larger, the performance of the developed or applied model increases.

5. Conclusions

The increase in application production in recent years may adversely affect the decision-making processes of users. Users find it difficult to choose the most suitable one from so many applications. Experimental results show that it is appropriate to use Ensemble Learning, one of the machine learning methods, for Google Play content rating analysis.

Ensemble Learning is preferred when classical classification algorithms are insufficient in machine learning problems. In this study, it has been observed that Ensemble Learning algorithms give better results than classical machine learning algorithms when estimating content rating.

When classical machine learning algorithms are applied to the content rating classification alone, KNN gives 79.9%, Artificial Neural Network 83.6%, and Logistic Regression 80.3%. However, when these classical machine learning models create an Ensemble Learning model with the Stacking algorithm, the classification success rises to 85.3%. Thus, bringing together more than one classifier model emerges as a useful process for content rating classification. Adaboost and Bagging algorithms showed higher results than Logistic Regression and KNN algorithms.

The increasing number of applications on Google Play makes it difficult for parents to follow their children. With this huge increase, a decision support mechanism should be created for the content rating process. Content rating classification and modeling of this classification are needed for the decision support mechanism. This study has made a Content Rating classification based on Community Learning. In this way, it is ensured that the applications in the marketplace are categorized by age group.

When Ensemble Learning algorithms are applied to a single data set, a higher success rate is expected rather than a 0.5% success rate. The increase in success should be more noticeable because Ensemble Learning algorithms combine multiple weak learners (classifiers or regressors). While the success of classification with traditional methods is 79% and 80%, this study increases it to 85% and 86% with Ensemble Learning algorithms.

Authors' Contributions

EA and TT provided the data and made it available for analysis, modeling, and forecasting. SB determined the methods to be used and the metrics to be used for theoretical calculations. EA, TT, and SB co-authored the data analysis and classification processes. SB is also the overall supervisor of the study.

All authors read and approved the final manuscript.

Competing Interests

The authors declare that they have no competing interests.

References

[1]. Ali, S., Elgharabawy, M., Duchaussoy, Q., Mannan, M., and Youssef, A., "Betrayed by the Guardian: Security and Privacy risks of parental control solutions". In Annual Computer Security Applications Conference, (pp. 69-83), (2020).

[2]. Alelyani, T., Ghosh, A. K., Moralez, L., Guha, S., and Wisniewski, P. "Examining parent versus child reviews of parental control apps on Google Play". In International Conference on Human-Computer Interaction, pp. 3-21. Springer, Cham, (2019).

[3]. Maredia, R. "Analysis of Google Play Store Data set and predict the popularity of an app on Google Play Store".

[4]. Wang, H., Li, H., Li, L., Guo, Y., and Xu, G. "Why are android apps removed from google play? a large-scale empirical study", In 2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR), pp. 231-242. IEEE, (2018).

[5]. Mueez, A., Ahmed, K., Islam, T., and Iqbal, W. "Exploratory data analysis and success prediction of Google Play Store app", (BRAC University), (2018).

[6]. Kılınç, M., Tarhan, Ç., and Aydın, C. "Could Mobile Applications' Success be Increased via Machine Learning and Business Intelligence Methods?" Avrupa Bilim ve Teknoloji Dergisi, (2020), (20), 805-814.

[7]. Sadiq, S., Umer, M., Ullah, S., Mirjalili, S., Rupapara, V., and Nappi, M. "Discrepancy detection between actual user reviews and numeric ratings of Google App store using deep learning." Expert Systems with Applications, 2021, 181, 115111.

[8]. Umer, M., Ashraf, I., Mehmood, A., Ullah, S., and Choi, G. S. "Predicting numeric ratings for Google apps using text features and ensemble learning", ETRI Journal, 2021, 43(1), 95-108.

[9]. Bashir, G. M. M., Hossen, M. S., Karmoker, D., and Kamal, M. J. "Android apps success prediction before uploading on google play store", In 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI) (pp. 1-6). IEEE, (2019).

[10]. AmanUllah, H., Fatima, M., Muneer, U., Ilyas, S., Rehman, R. A., and Afzal, I. "Causal Impact Analysis on Android Market", International Journal of Advanced Computer Science and Applications, 2019, 10(6).

[11]. Garg, M., Monga, A., Bhatt, P., and Arora, A. "Android app behavior classification using topic modeling techniques and outlier detection using app permissions", In 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC) (pp. 500-506). IEEE, (2016).

[12]. Magar, B. T., Mali, S., and Abdelfattah, E. "App Success Classification Using Machine Learning Models", In 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0642-0647), IEEE, (2021).

[13]. Shaw, E., Shaw, A., and Umphress, D. "Mining android apps to predict market ratings", In 6th International Conference on Mobile Computing, Applications and Services (pp. 166-167). IEEE, (2014).

[14]. Kaboha, N., Bani Hani, J., Seigneur, J. M., and Choukou, M. A. "The Role of Technology in Senior Co-Caregiving Support: A Scoping Review of Senior Care Mobile Applications", In 12th Augmented Human International Conference (pp. 1-2), (2021).

[15]. Ahmed, I., Ahmad, N. S., Ali, S., Ali, S., George, A., Danish, H. S., ... and Darzi, A. "Medication adherence apps: review and content analysis", JMIR mHealth and uHealth, 2018, (3), e6432.
[16]. Sambhi, R. D., Kalaichandran, R., and Tan, J. "Critical analysis of features and quality of applications for clinical management of acne", Dermatology online journal, 2019, 25(10).
[17]. Savic, M., Best, D., Rodda, S., and Lubman, D. I. "Exploring the focus and experiences of smartphone applications for addiction recovery", Journal of addictive diseases, 2013, 32(3), 310-319.
[18]. Krishnan, G., and Selvam, G. "Factors influencing the download of mobile health apps: Content review-led regression analysis", Health Policy and Technology, 2019, 8(4), 356-364.
[19]. Biviji, R., Vest, J. R., Dixon, B. E., Cullen, T., and Harle, C. A. "Factors related to user ratings and user downloads of mobile apps for maternal and infant health: Cross-sectional study", JMIR mHealth and uHealth, 2020, 8(1), e15663.
[20]. Ayyaswami, V., Padmanabhan, D. L., Crihalmeanu, T., Thelmo, F., Prabhu, A. V., and Magnani, J. W. "Mobile health applications for atrial fibrillation: a readability and quality assessment", International journal of cardiology, 2019, 293, 288-293.
[21]. Chyjek, K., Farag, S., and Chen, K. T. "Rating pregnancy wheel applications using the APPLICATIONS scoring system", Obstetrics & Gynecology, 2015,125(6), 1478-1483.
[22]. Frie, K., Hartmann-Boyce, J., Jebb, S., Albury, C., Nourse, R., and Aveyard, P. "Insights from Google Play Store User Reviews for the Development of Weight Loss Apps: An App Market Review", JMIR mHealth and uHealth, 2017, 5(12).
[23]. Takawale, H. C., and Thakur, A. "Talos app: On-device machine learning using TensorFlow to detect Android malware", In 2018 Fifth International Conference on Internet of Things: Systems, Management, and Security (pp. 250-255). IEEE, (2018).
[24]. Garg, S., and Baliyan, N. "Data on vulnerability detection in android", Data in brief, 2019, 22, 1081-1087.
[25]. Mealings, K., and Beach, E. F. "A content analysis of behavior change techniques in noise monitoring apps", Hear. Heal. Technol. Matters, 2020, pp. 1-30.
[26]. Siddiqui, N. R., Hodges, S., and Sharif, M. O. "Availability of orthodontic smartphone apps", Journal of orthodontics, 2019, 46(3), 235-241.
[27]. McIlroy, S., Ali, N., Khalid, H., and Hassan, A. E. "Analyzing and automatically labeling the types of user issues that are raised in mobile app reviews", Empirical Software Engineering, 2016, 21(3), 1067-1106.
[28]. Kishore Kolakaluri, D. R. and Mooramreddy Sreedevi. "Classification Of Google Playstore Apps Using Knn & Svm", 2020, 13(8), 183-190.
[29]. Meacham, M. C., Vogel, E. A., and Thrul, J. "Vaping-Related Mobile Apps Available in the Google Play Store After the Apple Ban: Content Review", Journal of medical Internet research, 2020, 22(11), e20009.
[30]. Mahmood, A. "Identifying the influence of various factors of apps on google play apps ratings", Journal of Data, Information, and Management, 2020, 2(1), 15-23.
[31]. Malavolta, I., Ruberto, S., Soru, T., and Terragni, V. "Hybrid mobile apps in the google play store: An exploratory investigation", In 2015 2nd ACM international conference on mobile software engineering and systems, (pp. 56-59), IEEE, (2015).
[32]. Sagi, O., and Rokach, L. "Ensemble learning: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2018, 8(4), e1249.
[33]. Onan, A. "Haber Metinlerinden Sosyo-ekonomik ve Epidemiyolojik Konuların Metin Madenciliğine Dayalı Belirlenmesi", Avrupa Bilim ve Teknoloji Dergisi, 2021, (26), 295-300.
[34]. Schapire, R. E. "Explaining AdaBoost. In Empirical inference", (pp. 37-52). Springer, Berlin, Heidelberg, (2013).
[35]. Freund Y, Schapire RE. "A Decision-theoretic generalization of online learning and an application to boosting". Journal of Computer and System Sciences, 1997, 55(1), 119-139.

[36]. Kalaycı, T. E. “Kimlik hırsızı web sitelerinin sınıflandırılması için makine öğrenmesi yöntemlerinin karşılaştırılması”, Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 2018, 24(5), 870-878.
[37]. Breiman, L., “Bagging predictors”, Machine learning, 1996, 24(2), 123-140.
[38]. Schwenk, H., and Bengio, Y. “Training methods for adaptive boosting of neural networks for character recognition”, Advances in neural information processing systems, 1998, 10, 647-653.
[39]. Breiman, L. “Random forests”, Machine learning, 2001, 45(1), 5-32
[40]. SİNGH, A. “Classification of Malware in HTTPS Traffic Using Machine Learning Approach”, El-Cezeri, 2022, 9(2), 644-655.
[41]. YARĞI, V., ve POSTALCIOĞLU, S. “EEG işareti kullanılarak bağımlılığa yatkınlığın makine öğrenmesi teknikleri ile analizi”, El-Cezeri, 2021, 8(1), 142-154.
[42]. Celil, O., ve DENER, M. Makine Öğrenme Metotları Kullanılarak KSA Ddos Saldırıları Tespiti. El-Cezeri, 2021, 8(3), 1550-1564.
[43]. Wikipedia. (2021, July 1). RandomForest. [Online].Available: https://en.wikipedia.org/wiki/Randomforest.
[44]. Timuçin, T., and Argun, İ. D. “Initial Seed Value Effectiveness on Performances of Data Mining Algorithms”, Düzce Üniversitesi Bilim ve Teknoloji Dergisi, 2021, 9(2), 555-567.
[45]. Doğaner, A., and Kirişçi, M. “CLASSIFICATION OF CORONARY ARTERY DISEASES USING STACKING ENSEMBLE LEARNING METHOD”, The Journal of Cognitive Systems, 2020, 5(2), 69-73.
[46]. Lanes, M., Schiavo, P. F., Pereira Jr, S. F., Borges, E. N., and Galante, R. “An Analysis of the Impact of Diversity on Stacking Supervised Classifiers”, In ICEIS (1), (pp. 233-240), (2017).
[47]. Cover, T., and Hart, P. “Nearest neighbor pattern classification”, IEEE transactions on information theory, 1967,13(1), 21-27.
[48]. GÜNEN, M. A. “Nokta Bulutu Verisi Kullanılarak Elma Bahçesinden Meyve Tespiti. El-Cezeri, 2022, 9(1), 253-265.
[49]. Kaya, D., Türk, M., and Kaya, T. “Examining the effect of dimension reduction on EEG signals by k-nearest neighbors algorithm”, El-Cezerî J. Sci. Eng, 2018, 5, 591-595.
[50]. Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., and Klein, M. “Logistic regression”, New York: Springer-Verlag, (2002).
[51]. Basheer, I. A., and Hajmeer, M. “Artificial neural networks: fundamentals, computing, design, and application”, Journal of microbiological methods, 2000, 43(1), 3-31.
[52]. Lavanya. (2021, July 10). Google Play Store Apps [Online]. Available: https://www.kaggle.com/lava18/google-play-store-apps.