Journal of Information System, Graphics, Hospitality and Technology
Volume 4, Nomor 1, Maret 2022
P-ISSN : 2622-1594
E-ISSN : 2685-449X

Analisa Klasifikasi Genre Game PC Terpopuler

Muhammad Rivaldy Hisham*, Jumiliono Pratama, Luky Andito, Andy Kho, Hendry Wijaya
Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Internasional Batam
E-mail: 1831032.muhammad@uib.edu, 1831090.jumilliono@uib.edu, 1831106.luky@uib.edu, 1831089.andy@uib.edu, 1831113.hendry@uib.edu

Abstrak— ***Kemajuan teknologi yang terus berkembang dengan pesat memungkinkan banyak perusahaan memanfaatkan teknologi dengan menciptakan berbagai macam cara dengan menggunakan sistem serta aplikasi dari Teknologi Informasi, salah satu contoh dari sistem TI ialah video game. Maraknya aplikasi video game dengan berbagai kategori yang telah dirancang dan diimplementasikan lalu dipublikasikan pada platform Google Play Store sangat memungkinkan penggunanya memberikan penilaian dan akan berdampak pada peringkat video game yang dipublikasikan. Pada penelitian ini, penulis menggunakan dataset Google Play Store yang diperoleh dari situs website Kaggle untuk memprediksi aplikasi yang populer menggunakan dua model klasifikasi yaitu Random Forest Classifier (RFC) dan Gradient Boosting Decision (GBD) dan membandingkan akurasi model ini. Penelitian ini menghasilkan prediksi aplikasi manakah yang populer dan tidak menggunakan dua model serta menentukan kategori video gamenya. Penelitian ini diharapkan dapat membantu perusahaan mempertimbangkan aplikasi atau video game apa yang baik untuk dikembangkan pada masa yang akan datang.***

Kata Kunci— Gradient Boosting Decicision, Random Forest Classifier, Klasifikasi, Video Game

I. PENDAHULUAN

Perkembangan aplikasi *mobile* saat ini begitu pesat, hal ini di akibatkan karena jumlah pengguna *smartphone* yang meningkat dratis. Per 20 Januari tahun 2020 saja dicatat setidaknya ada sekitar 3,2 miliar pengguna dan juga perangkat yang aktif sekitar 3,8 miliar unit [1]. *Google Play Store* merupakan distribusi layanan digital yang dikembangkan oleh *Google* dan pertama kali di luncurkan pada tanggal 6 Maret tahun 2012, yang bertujuan untuk menyatukan pasar *Android, Google Music* dan *Google e-Bookstore* dalam satu aplikasi. Sehingga sekarang *Google Play Store* menjadi toko aplikasi resmi untuk sistem operasi *Android* yang di dalamnya kita dapat menginstal aplikasi seperti toko digital, musik, film, buku, maupun *video game*. *Google Play Store* mempunyai beragam aplikasi termasuk yang berbayar ataupun gratis untuk pengguna *Android*.

Naskah Masuk : 10 Mei 2021
Naskah Direvisi : 28 Desember 2021
Naskah Diterima : 19 Januari 2022
*Corresponding Author : 1831032.muhammad@uib.edu

Saat tahun 2016 silam, aplikasi yang terunduh pada GPS sebanyak 82 miliar, lebih dari 3,5 juta aplikasi yang sudah dipublikasikan di GPS pada tahun 2017, sehingga banyak perusahaan yang melakukan pengembangan pada operasi sistem *Android* dan salah satunya adalah *video game*. Peningkatan pengguna dalam video game juga terlihat pada situasi pandemi COVID-19 dikarenakan kita diharuskan untuk melakukan *social distancing*[2].

Setiap perusahaan berusaha untuk menghasilkan *video game* yang berkualitas, sehingga perusahaan akan mendapatkan *profit* dari *video game* yang sudah dipublikasikan. Keuntungan dari aplikasi video game bukan hanya dari iklan yang ditayangkan tetapi juga dari pembelian dari dalam aplikasi dan/atau *video game* tersebut. Ada beberapa faktor yang dapat mepengaruhi keseruan bermain *game*, yaitu : keterhubungan, penghargaan, fleksibilitas akses dan peringkat [3].

Game yang berkualitas tentu saja tidak hanya dilihat dari sisi desain ataupun grafik tetapi masih ada faktor lain seperti suara, alur cerita, ataupun *gameplay* yang merupakan kebutuhan untuk meningkatkan kualitas game tersebut [4]. Banyaknya jumlah *game* yang terah dirilis oleh GPS membuat penulis ingin melakukan penelitian *video game* dengan genre apa yang populer di GPS. Pada penelitian sebelumnya, telah dilakukan klasifikasi yang bertujuan untuk memprediksi dan membandingkan tingkatan akurasi dan popularitas pada video game di GPS menggunakan beberapa algoritma [5] yaitu *Random Forest Classifier* (RFC) yang merupakan salah satu teknik pembelajaran asambel yang paling sukses. Teknik ini telah terbukti menjadi teknik yang sangat populer dalam pengenalan pola dan pembelajaran mesin untuk klasifikasi [6]. *Gradient Boosting Decision* (GBD) adalah algoritma *Machine Learning* yang menggunakan banyak *decision tree* sebagai pembelajar dasar. Setiap *decision tree* tidak independen dikarenakan *decision tree* baru yang ditambahkan meningkatkan sampel yang akan di klasifikasikan dari *decision tree* sebelumnya [7] dengan menggunakan algoritma RFC dan GBD untuk memprediksi kategori *game* apa yang sedang populer sehingga dengan adanya penelitian ini para pengembang dan *programmer* pihak *video game* dapat bisa mengetahui *video game* apa yang akan dikembangkan dalam masa yang akan datang.

DOI : 10.37823/insight.v4i01.145

II. METODE PENELITIAN

A. Data Understanding

Pada setiap penelitian, dibutuhkan suatu cara dan/atau metode untuk menyelesaikan sebuah permasalahan. Pada penelitian ini, dibutuhkan suatu metode yang dapat memandu penulis dalam pendekatan data sains yaitu metode *Cross-Industry Standard Process for Data Mining* (CRISP-DM) [8]. Terdapat beberapa tahapan proses yang perlu dijalankan agar penelitian ini dapat berjalan dengan baik, antara lain :

1. Business Understanding

Pada tahap ini penulis diminta melakukan pemahaman bisnis, yang dimana penulis perlu mengetahui tujuan, sasaran, dan juga urgensi dari permintaan tersebut. Setelah menganalisa kebutuhan bisnis, penulis perlu mengindetifikasi teknik yang cocok untuk mencapai hasil yang diinginkan, yang dimana hasil dari penelitian ini adalah menprediksi tipe kategori *video game* apa yang terpopuler.

2. Data Understanding

Pada tahap ini penulis akan menentukan data-data apa saja yang akan digunakan, karena data-data tersebut akan memiliki pengaruh pada algoritma yang digunakan nantinya. Penulis menggunakan *dataset* yang telah disediakan oleh website *Kaggle*[9] merupakan tempat set data yang sudah disediakan dan juga sudah terdapat daftar dari aplikasi yang sudah dipublikasikan pada GPS dikarenakan penelitian ini bertujuan untuk menganalisa tipe kategori *game* yang populer sehingga penulis hanya menggunakan beberapa atribut set data yang tersedia pada web tersebut.

3. Data Preparation

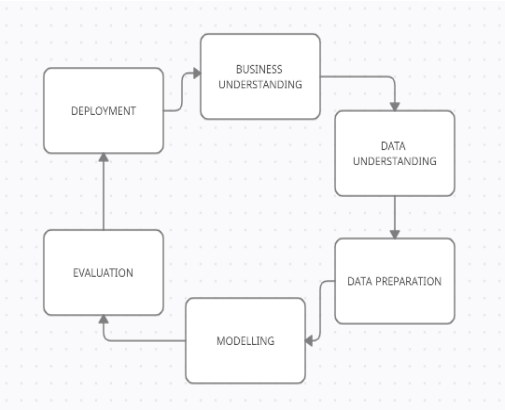
Pada tahap ini penulis akan mempersiapkan data untuk masuk kedalam tahap pemodelan, dimana penulis akan memisahkan beberapa atribut yang berupa kalimat dan menentukan atribut apa saja yang dapat digunakan untuk klasifikasi, data tersebut akan dibagi lagi menjadi 2 dataset yang dimana itu adalah dataset training dan test.

4. Modeling

Pada tahap ini penulis akan melakukan modeling terhadap dataset yang telah tersedia menggunakan algoritma *Random Forest Classifier* dan *Gradient Boosting Decision*, jika hasil prediksi yang dihasilkan kurang memuaskan. maka penulis akan kembali lagi ke tahap Data Preparation. Hasil dari tahap modeling ini akan berupa angka yang menentukan berapa tinggi tingkat keakurasian dalam memprediksi menggunakan algoritma tersebut.

5. Evaluation

Pada tahap yang terakhir penulis akan mengevaluasi dan membandingkan performa model yang telah dibuat.



Gambar 1. Flowchart CRISP-DM (diolah penulis)

B. Random Forest Decision

RFC merupakan kombinasi dari masing-masing *tree* yang kemudian dikombinasikan kedalam suatu model. RFC bergantung pada sebuah nilai *random* vector dengan nilai distribusi sama pada semua turunan yang mana masing-masing *decision tree* memiliki kedalaman yang maksimal. Hal ini banyak mengunakan algoritma agar kesederhanaan dan kemudahan dalam mengukur faktor prediksi menjadi mudah[10]. Model RFC bekerja pada konsep *corelated decision trees*. Untuk mendapatkan hasil yang akurat dan stabil, RFC membuat beberapa pohon dan menggabungkannya dalam suatu hutan acak keputusan masing-masing pohon dan membantu meningkatkan akurasi. Analisis penggunaan kembali perangkat lunak pendekatan menggunakan hutan acak *gradient boosting machi*ne (GBM) untuk meningkatkan akurasi. Formula ini menggunakan *bagging* untuk keacakan. *Bagging* adalah singkatan dari *bootstrap aggregating,* dimana akurasi dan stabilitas algoritma ditingkatkan secara matematis, dimana x′ adalah prediksi untuk sampel tak terlihat, b adalah jumlah pohon yaitu, b = 1,2,3… B; dan fb = Latih fb DT pada Xb, Yb.

$$bagging = \frac{1}{B}\sum_{b=1}^{B} \text{fb}(x')$$

C. Gradient Boosting Machine

GBM merupakan model algoritma percabangan dari RFC tetapi sementara algoritma GBM mampu menangani campuran tipe data, menghasilkan daya prediksi yang baik, dan kuat terhadap outlier (dengan fungsi loss yang kuat), mereka mungkin tidak diparalelkan karena sifat peningkatan yang berurutan[11].

III. HASIL DAN PEMBAHASAN

Penulis menggunakan dataset GPS yang telah tersedia pada web *Kaggle* dan didapatkanlah 17 atribut yang akan digunakan untuk memprediksi jumlah kategori *video game* terpopuler. Penulis memisahkan atribut yang tersedia dengan tidak menggunakan aplikasi akuntansi *Microsoft Excel* dan berikut adalah atribut yang akan digunakan dalam penelitian ini:

DOI : 10.37823/insight.v4i01.145

```
df_apps.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 960 entries, 0 to 959
Data columns (total 11 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Rank          960 non-null    int64
 1   Name          960 non-null    object
 2   Platform      960 non-null    object
 3   Year          943 non-null    float64
 4   Genre         960 non-null    object
 5   Publisher     954 non-null    object
 6   NA_Sales      960 non-null    float64
 7   EU_Sales      960 non-null    float64
 8   JP_Sales      960 non-null    float64
 9   Other_Sales   960 non-null    float64
 10  Global_Sales  960 non-null    float64
dtypes: float64(6), int64(1), object(4)
memory usage: 82.6+ KB
```

Gambar 2 : Atribut Dataset

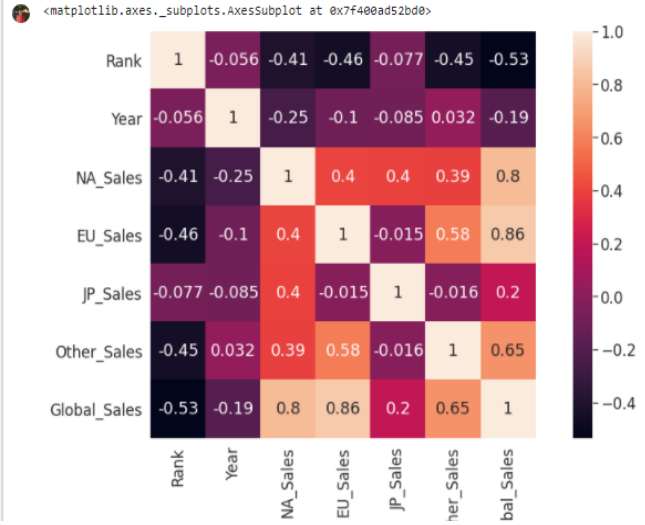
#	Kolom	*Non-Null Count*	Tipe Data
0	Ranking	*960 non-null*	*Int64*
1	Nama	*960 non-null*	Objek
2	Platform	*960 non-null*	Objek
3	Tahun	*960 non-null*	*Float64*
4	Genre	*960 non-null*	Objek
5	Penerbit	*960 non-null*	Objek
6	Penjualan Amerika Utara	*960 non-null*	*Float64*
7	Penjualan Uni Eropa	*960 non-null*	*Float64*
8	Penjualan Jepang	*960 non-null*	*Float64*
9	Penjualan yang lain	*960 non-null*	*Float64*
10	Penjualan Internasional	*960 non-null*	*Float64*

Tabel 1. Atribut Dataset

Correlation heatmap adalah hubungan antar variabel yang terdapat dalam set data, tidak semua variable dapat digunakan dalam visualisasi korelasi map panas, hanya variabel dataset numerik saja yang dapat digunakan. Salah nya pengambilan set data dapat mempengaruhi saat penelitian atau pembelajaran suatu algoritma [12].

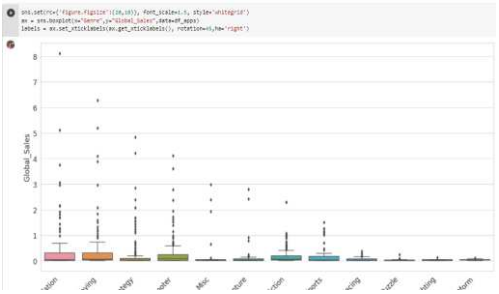
```
import matplotlib.pyplot as plt

num_col = df_apps.dtypes[df_apps.dtypes != 'object']

fig = plt.figure(figsize = (15,8))
sns.heatmap(df_apps[list(num_col.index)].corr(),annot = True,square = True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f400ad52bd0>
```



Gambar 3 : Correlation Heatmap

Kemudian terdapat *box-plot* untuk menuntukan hasil rata-rata rating dari keseluruhan aplikasi yang di group berdasarkan kategori. Dapat dilihat pada Gambar 4, terdapat beberapa kategori yang bar nya menyentuh angka 0 karena dataset yang didapatkan, aplikasi tersebut tidak diketahui jumlah ratingnya.

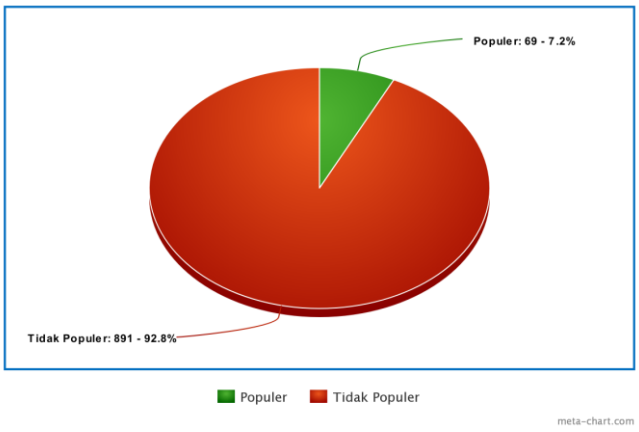


Gambar 4 : Box-plot Rating

A. *Data Preparation*

Pada tahap Data preparation penulis mengconvert atribut *price* dan *currency* ke bentuk USD, kemudian mengubah *size* ke kb, mengubah *column Varies with device* pada atribut *size* dan *Minimum Android* menjadi 0 meggunakan *Excel*. Kami mengklasifikasikan aplikasi populer dalam 2 nilai yaitu unpopuler (0) dan populer (1). Kami menetapkan bahwa aplikasi yang di install lebih dari 100 ribu kali akan ditetapkan sebagai aplikasi yang populer dan membagi data menjadi 80:20 yang dimana data training 80% dan data test 20%.

DOI : 10.37823/insight.v4i01.145



Gambar 5 : Populer dan tidak populer aplikasi

B. *Modelling RFC*

Random Forest merupakan algoritma yang populer dalam pengenal pola dan klasifikasi. Pertimbangkan pembelajaran *Random Forest* adalah L = ((M1, N1),.., (Mn, Nn)) yang dimana n vektor, M ∈ X dimana X himpunan observasi dalam bentuk numerik dan N ∈ Y dimana Y adalah himpunan kelasnya. Sistem pembelajaran menghasilkan pengklasifikasi dari *sample* dan menggabungkan semua pengklasifikasi yang dihasilkan dari uji coba yang berbeda untuk membentuk pengklasifikasi akhir. Kami mengset n_estimators sebesar 20 yang artinya terdapat 20 decision *tree* yang ada dalam proses *training*

C. *Modelling GBD*

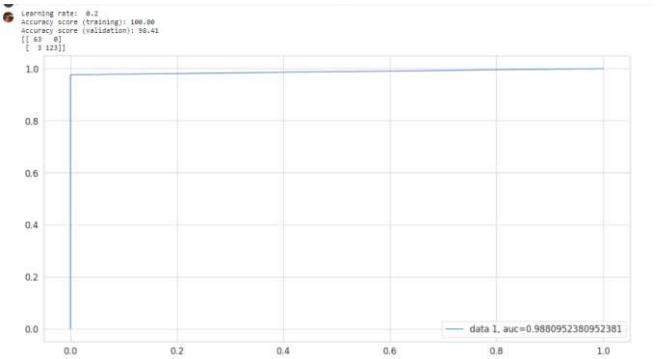
Gradient Boosting Decision merupakan algoritma yang dapat membangun *decision tree* berdasarkan peningkatan dari pohon pembelajaran yang lemah untuk memperbaiki kesalahan pohon dan mencegah terjadinya potensi *overfitting*. Algoritma ini juga mampu memecahkan masalah dengan menyesuaikan pembelajaran lemah dengan gradien negatif dari fungsi kerugian (*loss function*) dan meningkatkan pohon (*trees*) dengan parameter yang mewakili variabel split yang dipasang pada setiap node terminal pohon.

D. *Evaluation*

```
[[63  0]
 [47 79]]
              precision    recall  f1-score   support

           0       0.57      1.00      0.73        63
           1       1.00      0.63      0.77       126

    accuracy                           0.75       189
   macro avg       0.79      0.81      0.75       189
weighted avg       0.86      0.75      0.76       189

Accuracy : 75.13
```

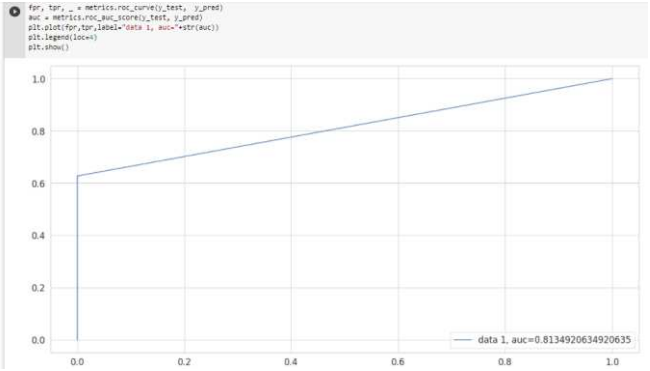
Gambar 6 : Confusion Matrix & Accuracy

Dari gambar diatas disimpulkan bahwa, penggunaan algoritma *Random Forest* dalam pengklasifikasian *game* populer sudah tepat, dengan tingkat akuarsi 100% dan jumlah data yang di training sebanyak 33089. Berdasarkan hasil dari confusion matrix diatas, hasil prediksi di baris pertama dengan kategori unpopuler sudah sesuai dengan dataset pada kolom pertama yang dilambangkan dengan 0. Pada baris kedua, hasil prediksi dengan kategori populer sudah sesuai dengan dataset pada kolom kedua yang dilambangkan dengan 1. *Receiver Operating Characteristics* (ROC) merupakan alat ukur *performance* untuk *classification* yang sering digunakan menentukan *threshold* dari suatu model.



Gambar 7 : Receiver Operating Characteristic Random Forest

Pada *classification report gradient boosting* tingkat keakurasian pada algoritma sebesar 94.9%. Berdasarkan hasil dari confusion matrix diatas, hasil prediksi di baris pertama dengan kategori unpopuler sudah sesuai dengan dataset pada kolom pertama yang dilambangkan dengan 0 terdapat sekitar 11000 data. Pada baris kedua kolom pertama, hasil prediksi dengan kategori populer tidak sesuai dengan dataset yaitu terdapat sekitar 2300 data. Pada baris kedua kolom kedua, hasil prediksi dengan kategori populer sudah sesuai dengan dataset pada kolom kedua yang dilambangkan dengan 1 yaitu terdapat sekitar 20000 data. Pada gambar 10, ditunjukkan bahwa dengan menggunakan algoritma gradient boosting dapat belajar, dengan melihat terjadi nya peningkatan pada gambar.



Gambar 8 : Receiver Operating Characteristic Gradient boosting Decision

Confusion Matrix & Accuracy

```
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score, recall_score
print(confusion_matrix(y_test,y_pred))
print(classification_report(y_test,y_pred))
print('Accuracy : %.2f' %(accuracy_score(y_test, y_pred.round())*100))

[[63  0]
 [47 79]]
              precision    recall  f1-score   support

           0       0.57      1.00      0.73        63
           1       1.00      0.63      0.77       126

    accuracy                           0.75       189
   macro avg       0.79      0.81      0.75       189
weighted avg       0.86      0.75      0.76       189

Accuracy : 75.13
```

Gambar 9 : Classification Report Gradient Boosting Decision

Kemudian penulis menentukan aplikasi yang populer dengan cara pemfilteran yang dimana rating diatas 4 rating count diatas 50 ribu, dan maximum install lebih dari 100 ribu. Kemudian data tersebut di *group* berdasarkan *category* dan menggunakan *function count* untuk menghitung jumlah aplikasi pada *category* tersebut. Dari pemfilteran menunjukan bahwa aplikasi yang paling banyak adalah Puzzle.

DOI : 10.37823/insight.v4i01.145

```
highRating = df_apps.copy()
highRating = highRating.loc[highRating["Global_Sales"] > 1]
highRateNum = highRating.groupby('Genre')['Global_Sales'].nunique()
highRateNum
```

```
Genre
Action          4
Adventure       2
Misc            3
Role-Playing    13
Shooter         10
Simulation      15
Sports          6
Strategy        14
Name: Global_Sales, dtype: int64
```

Gambar 10 : Filter Populer Game

IV. KESIMPULAN

Berdasarkan pengujian dan analisis dari klasifikasi game populer di *Google Playstore* menggunakan model *Random Forest Classifier* dan *Gradient Boosting Decision,* maka kita dapat menyimpulkan bahwa:

a. Dengan menggunakan algoritma *Random Forest Classifier* dan *Gradient Boosting Decision* penulis berhasil melakukan pengklasifikasian aplikasi *video game* populer.

b. Dari 17 *category game* dalam dataset, aplikasi yang rating lebih dari 4 dengan jumlah terbanyak adalah Puzzle.

c. Pada algoritma *Gradient Boosting Decision* terjadi *overfitting*. *Overfitting* adalah keadaannya data digunakan untuk pelatihan adalah yang terbaik, sehingga saat dilakukan *model training* menggunakan data yang berbeda dapat mengurangi tingkat keakurasian. hal ini dapat diatasi dengan meningkatkan *learning rate* pada proses *training* yang dimana default nya adalah 0.1.

d. Dengan tinggi nya tingkat keakurasian yang didapatkan dari kedua model diatas, menandakan kita dapat menentukan aplikasi yang populer berdasarkan dari jumlah *maximum installs* dari aplikasi tersebut.

DAFTAR PUSTAKA

[1]Y. Pusparisa, “Berapa Jumlah Pengguna Smartphone Dunia,”*20-01-2020*, 2019. .

[2]D. L. King, P. H. Delfabbro, J. Billieux, and M. N. Potenza, “Problematic Online Gaming and The COVID-19 Pandemic,”*J. Behav. Addict.*, vol. 9, no. 2, pp. 184–186, 2020, doi: 10.1556/2006.2020.00016.

[3]K. L. Hsiao and C. C. Chen, “What drives in-app purchase intention for mobile games? An examination of perceived values and loyalty,”*Electron. Commer. Res. Appl.*, vol. 16, pp. 18–29, 2016, doi: 10.1016/j.elerap.2016.01.001.

[4]A. Trisnadoli, “Jurnal Politeknik Caltex Riau Analisis Kebutuhan Kualitas Perangkat Lunak Pada Software Game Berbasis Mobile,”*Anal. Kebutuhan Kualitas Perangkat Lunak Pada Softw. Game Berbas. Mob.*, vol. 1, no. 2, pp. 67–74, 2015.

[5]R. Maredia, “Analysis of Google Play Store Data set and Predict The Popularity of An App On Google Play Store Analysis of Google Play Store Data Set and Pedict The Popularity of An App On Google Play Store,” no. June, pp. 1–6, 2020.

[6]Y. L. Pavlov,*Random forests*. 2019.

[7]J. H. Friedman, “Greedy function approximation: A gradient boosting machine,”*Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.

[8]S. Huber, H. Wiemer, D. Schneider, and S. Ihlenfeldt, “DMME: Data mining methodology for engineering applications - A holistic extension to the CRISP-DM model,”*Procedia CIRP*, vol. 79, pp. 403–408, 2019, doi: 10.1016/j.procir.2019.02.106.

[9]M. Carpita, E. Ciavolino, and P. Pasca, “Exploring and modelling team performances of the Kaggle European Soccer database,”*Stat. Modelling*, vol. 19, no. 1, pp. 74–101, 2019, doi: 10.1177/1471082X18810971.

[10]A. K. Sandhu and R. S. Batth, “Software reuse analytics using integrated random forest and gradient boosting machine learning algorithm,”*Softw. - Pract. Exp.*, vol. 51, no. 4, pp. 735–747, 2021, doi: 10.1002/spe.2921.

[11]A. Kadiyala and A. Kumar, “Applications of python to evaluate the performance of decision tree-based boosting algorithms,”*Environ. Prog. Sustain. Energy*, vol. 37, no. 2, pp. 618–623, 2018, doi: 10.1002/ep.12888.

[12]M. Chakradar, A. Aggarwal, and R. Forests, “FEATURE SELECTION FOR INSULIN RESISTANCE USING,” vol. 18, no. 04, pp. 4861–4879, 2021.

DOI : 10.37823/insight.v4i01.145