Your Future App

Abstract

This project aims to help new developers to decide what App types and characters people prefer. The goal was achieved by making Exploratory Data Analysis (EDA) and prediction models on the Google Play Store Apps dataset. I worked on a dataset found through the Kaggle website. I used python libraries such as NumPy, pandas, Sklearn, and Matplotlib.

Design

The data has been collected by using Python script in June 2021. This dataset presents 5 classes: **Too low installed, Low installed**, **Medium installed**, **Highly installed**, and **Too highly installed** for each App depending on the number of Installs. By Applying EDA the following questions have been answered:

- Which category has more Apps?
- To which Installs label the maximum installs Apps belongs?
- Who is the developer who has the maximum number of Apps?
- Will the price affect number of installations?
- What is the most downloaded app?
- What is the most famous category of the app?

Data

The dataset contains over 2 million instances with 22 features for each. Features include numerical and categorical types, such as rating, type, developer, etc. I have created two datasets from the original: the first by dropping all the rows that have NA values, the second by applying some handle missings techniques.

Algorithms

- Feature Engineering
  - Factorizing ContentRating, Category, and Installs columns, into integers values.
  - Converting True and False to 1 and 0 in Editors Choice, Ad-Supported, In-App Purchases, and Free columns.
  - Handle missing values in Minimum Installs, ContentRating, App Name, and Developer Id.
  - Drop some non-useful columns such as Currency, Size, and developer Email.

- Models
  - Classification model to classify App Installs.
  - Regression model to predict the rate of future App.
    By using Logistic Regression, Decision Tree, and Random Forest classifiers. The dataset has dived into 80% training and 20% testing. The models work with 16 features at the end. The scores used are Accuracy, F1, Precision, Recall, and RMSE. The Decision Tree was the best.

Tools

- Numpy and Pandas to handle the data.
- Matplotlib and Seaborn to visualize the data.
- Sklearn to build the models
- Google Colab to run the code.

Communication

Slides show and python graph to represent the results.