INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

Statistical Methods in AI

Course Project Report

Project Id : 39

“Smart Phone Prediction”

Team Id : 12

Anurag Chaturvedi (2018201024)

Neelesh Bhakt (2018201022)

Shivam Singh (2018201015)

Tarpit Sahu (2018201089)

Assigned TA

Vachaspathi Ramakrishnan

Introduction to The Project

Project Id :	39
Project Title :	Smartphone Predictor
Project Description :	Given various attributes of smart-phones, predict brand and model
Expectation :	Scrape web data to obtain features
Github Repository:	https://github.com/tarpitsahu/Smartphone-Predictor

Brief Introduction

The emergence of Artificial intelligence (AI) has led to applications which are now having a profound impact on our lives. This is a technology which is barely 60 years old. One of the popular applications of AI is Machine Learning, in which computers, software, and devices perform via cognition (similar to human brain).

Smart Phone prediction is one such application of Machine Learning which aims at simplifying decision making for a person interested in buying a smart-phone. The basic functioning of the project is as follows : user gives the input features of smart-phone, the model then predicts the phone which is closest to the input query.

The project is divided in ***4*** major sub-modules :

1. Data Collection

There is very less data (in structured form) present on the internet. Therefore, it was necessary to design our own dataset in order to proceed further. This phase deals about the methods used to collect the data.

2. Data Cleaning and Visualization

Data present on internet is in raw form which serves no purpose for our model. It is necessary to convert that raw data into structured form. Thsi phase deals about the steps taken in cleaning and visualization phases.

3. Feature Selection

For each smartphone, we collected values of over 30 features. Not all features are equally important. In this phase we applied different techniques for doing feature selection. Some of them are mentioned below:

a) *Feature Selection using Random Forest Algorithm*

b) *Feature Selection using Importance Value*

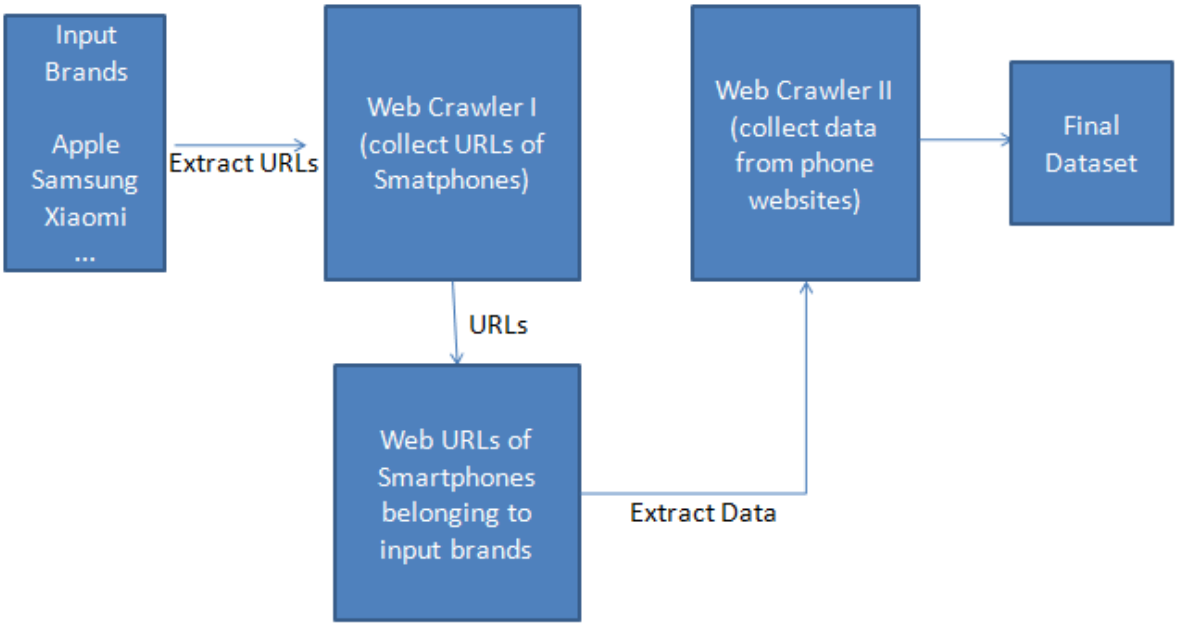
c) *Feature Selection using Correlation Matrix and Heat Maps.*

4. Algorithm Design

After completing above three steps, we have well structured data, ready to be fed into a machine learning algorithm. This phase deals with study undertaken to select best algorithm for the project.

Data Collection

Following Flow Chart illustrates the process adopted in order to collect the dataset :



Data Collection

We chose www.gsmarena.com website to collect information about various Smart-phones. The website was chosen because it contains data about almost every mobile phone, plus the structure of web page is apt for crawling.

We developed a web-crawler written in Python which is capable of reading the source code of the webpages and extract out relevant information from it.

Dataset Links

- https://drive.google.com/open?id=12iwxvYum8C5O2MCkR5DsEQBWMxuytrH5
- https://drive.google.com/open?id=1sp_M2ZAFUU84ut5WsFTHNn7cEetWKVVn

Following features for each smartphone were collected :

- **Model Name** : Brand and Name of the smartphone
- **Network Technology** : GSM, CDMA etc.
- **Year** : Year of release
- **Dimensions** : Length x Breadth x Depth
- **Weight** : Weight of the smartphone
- **Sim** : Number of SIM Cards
- **Display Type** : TFT, LED, Capacitive etc.
- **Display Size** : Size of the display 5”,6” etc
- **Display Resolution**
- **Operating System** : Android, Windows etc.
- **Chipset**
- **Central Processing Unit**
- **Graphical Processing Unit**
- **External Memory Slots**
- **Internal Memory**
- **GPS**
- **USB**
- **Battery Description**
- **NFC**
- **WLan**
- **Radio**
- **Sensors**

Our dataset consists of features of over ***700*** smart-phones of 6 brands viz. Apple, Samsung, Xiaomi, Sony, Honor, Motorola. Exact numbers are presented in the table below

Brand	Number Of Smart-phones
Samsung	308
Apple	58
Sony	113
Xiaomi	85
Motorola	107
Honor	52

Data Pre-Processing and Visualization

The data present on internet is in raw and unstructured format. Therefore it is necessary to clean and preprocess the data in order to suit our model.

- Formatting data in standard way
- Handling Missing Values
- Encoding data to convert into numerics

Feature Selection

Feature Selection is one of the core concepts in machine learning which hugely impacts the performance of your model. The data features that you use to train your machine learning models have a huge influence on the performance you can achieve.

Irrelevant or partially relevant features can negatively impact model performance. Feature selection and Data cleaning should be the first and most important step of your model designing.

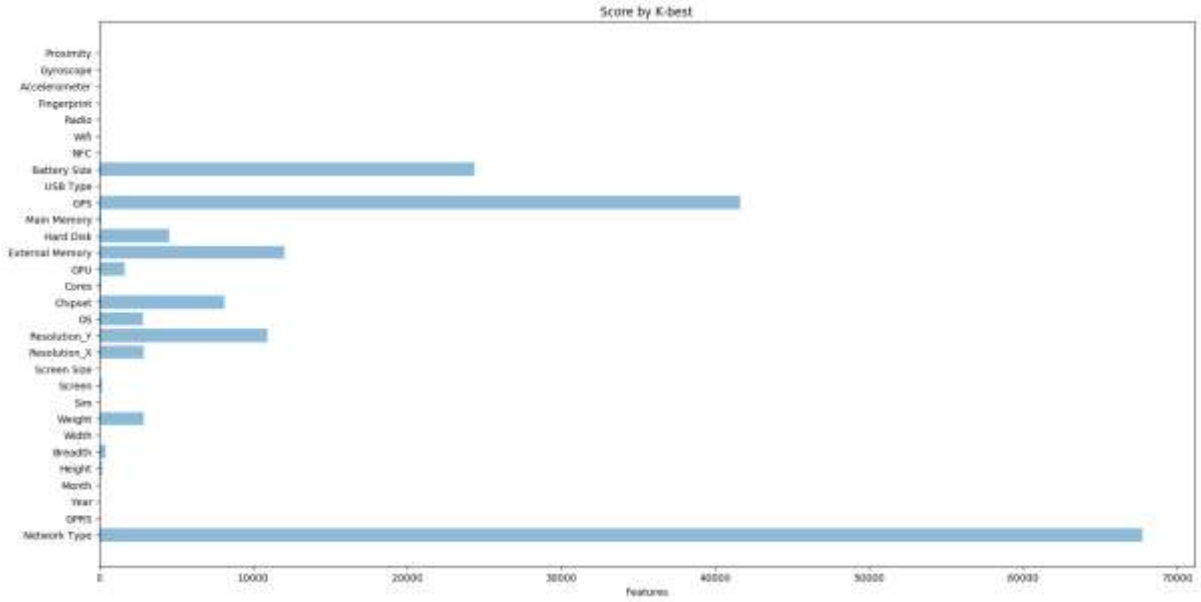
We performed following ***three*** ways of Feature Selection :

1. Uni-variate Selection
2. Feature Importance
3. Correlation Matrix with Heatmap

1. Univariate Selection

Statistical tests can be used to select those features that have the strongest relationship with the output variable.

The scikit-learn library provides the SelectKBest class that can be used with a suite of different statistical tests to select a specific number of features.

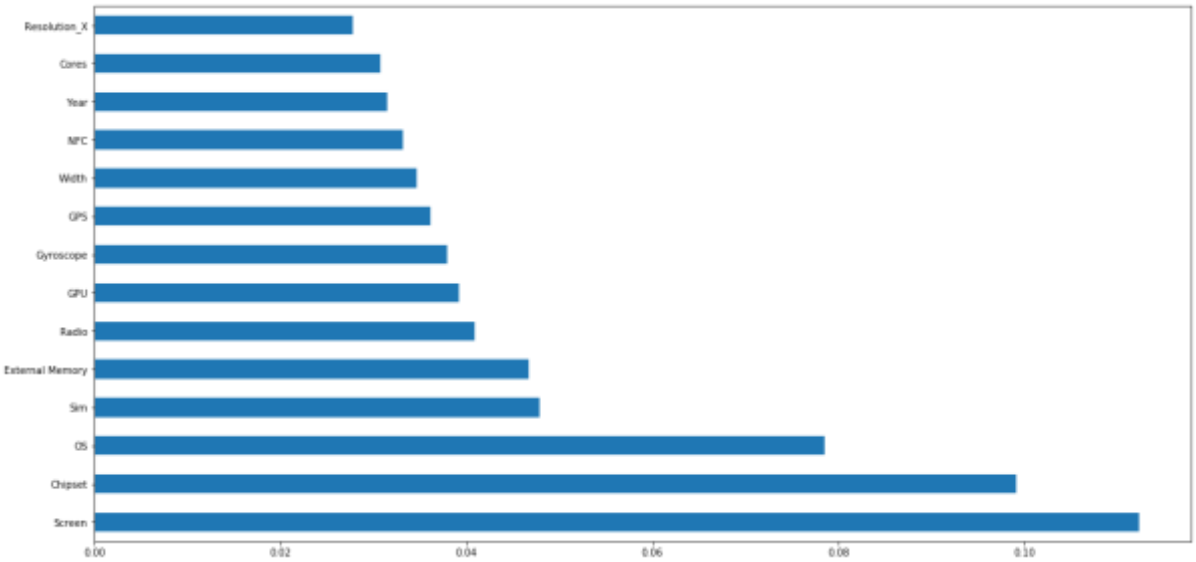


2. Feature Importance

You can get the feature importance of each feature of your dataset by using the feature importance property of the model.

Feature importance gives you a score for each feature of your data, the higher the score more important or relevant is the feature towards your output variable.

Feature importance is an inbuilt class that comes with Tree Based Classifiers, we will be using Extra Tree Classifier for extracting the top 14 features for the dataset.

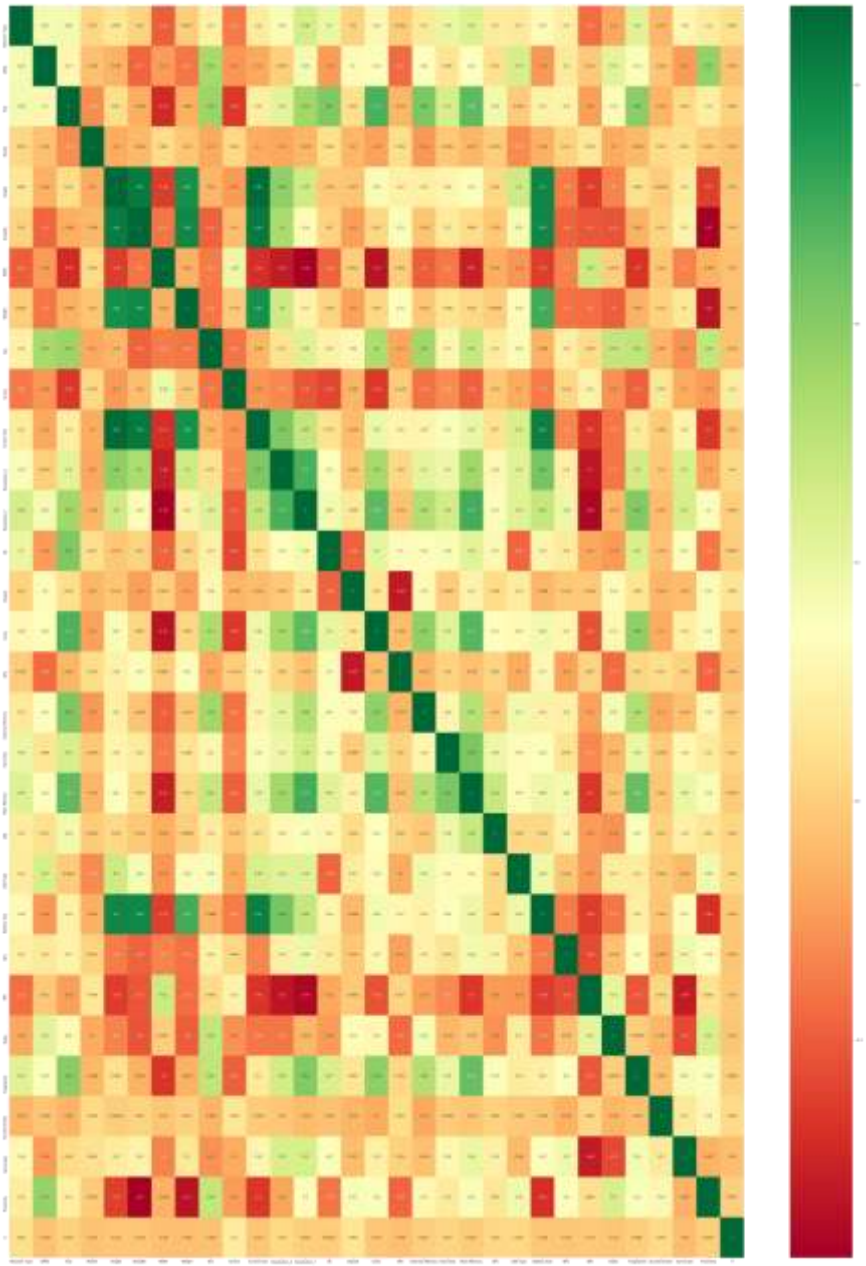


3. Correlation Matrix with Heatmap

Correlation states how the features are related to each other or the target variable.

Correlation can be positive (increase in one value of feature increases the value of the target variable) or negative (increase in one value of feature decreases the value of the target variable)

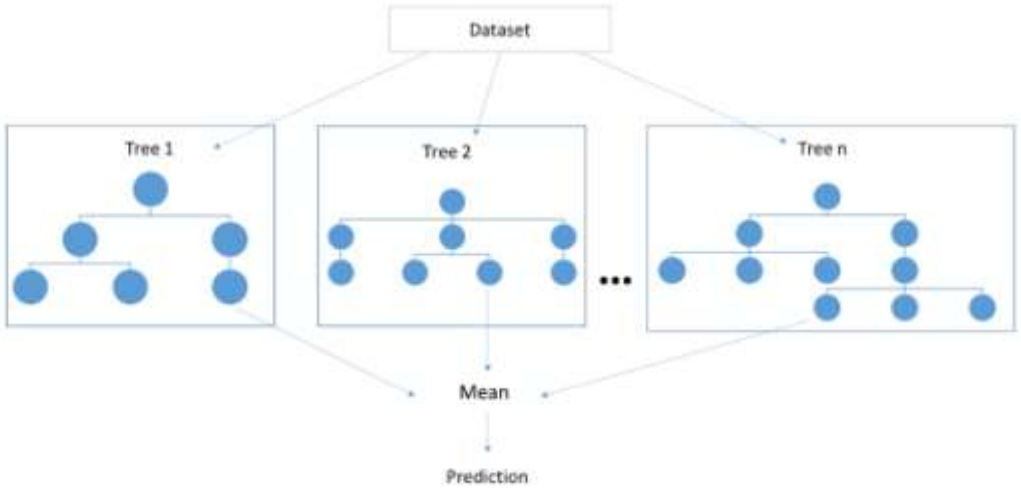
Heatmap makes it easy to identify which features are most related to the target variable, we will plot heatmap of correlated features using the seaborn library.



Algorithm Design

Random Forest Classifier

- It is ensemble learning method for classification.
- It constructs multiple decision trees in training, and gives the class that is the mode of the classes of the individual trees.
- It corrects overfitting.
- Trees that are grown very deep tend to learn highly irregular patterns: they overfit their training sets, i.e. have low bias, but very high variance. Random forests are a way of averaging multiple deep decision trees, trained on different parts of the same training set, with the goal of reducing the variance. This comes at the expense of a small increase in the bias and some loss of interpretability, but generally greatly boosts the performance in the final model.



Extra Tree Classifier

Adding one step further in randomization yields ExtraTrees.

This is also ab ensemble method, similar to random forests.

There are two difference in these two methods :

1) In Extra Trees, each tree is trained using the whole learning sample (rather than a bootstrap sample)

2) In Extra Trees, the top-down splitting is randomized.

Instead of computing the locally optimal cut-point for each feature under consideration, a random cut-point is selected.

This value is selected from a uniform distribution within the feature's empirical range (in the tree's training set). Then, of all the randomly generated splits, the split that yields the highest score is chosen to split the node. Similar to ordinary random forests, the number of randomly selected features to be considered at each node can be specified. Default values for this parameter are sqrt(n) for classification and n {\displaystyle n} n for regression, where n n is the number of features in the model.

Work Performed

Following cases were considered

EXPERIMENT 1.

Given Smartphone Features, Predict The Brand of Mobile Phone

In this experiment, a user will provide the features of a smartphone, then our model will output which brand the mobile in user query belongs to.

Classifier : Random Forest Classifier

Accuracy	67%
Precision	58%
Recall	66.2%
F1-Score	61.7%

Classifier : Extra Tree Classifier

Accuracy	55.6%
Precision	56.3%
Recall	51.7%
F1-Score	52.9%

Classifier : SVM Classifier

Accuracy	31.8%
Precision	5.3%
Recall	16.66%
F1-Score	8.0%

Classifier : KNN Classifier

Accuracy	42.15%
Precision	42.16%
Recall	38.63%
F1-Score	38.06%

Classifier : Naive Bayes Classifier

Accuracy	39.91%
Precision	57.10%
Recall	40.91%
F1-Score	39.35%

EXPERIMENT 2.

Given Smartphone Features, Predict The Brand+Model of Mobile Phone

In this experiment, a user will provide the features of a smartphone, then our model will output which brand the mobile in user query belongs to. In addition to this, It will also output what Model the user is talking about.

Classifier : Random Forest Classifier

Accuracy	40.30%
Precision	25.11%
Recall	20.26%
F1-Score	22.42%

Classifier : Extra Tree Classifier

Accuracy	27.45%
Precision	15.60%
Recall	16.22%
F1-Score	15.90%

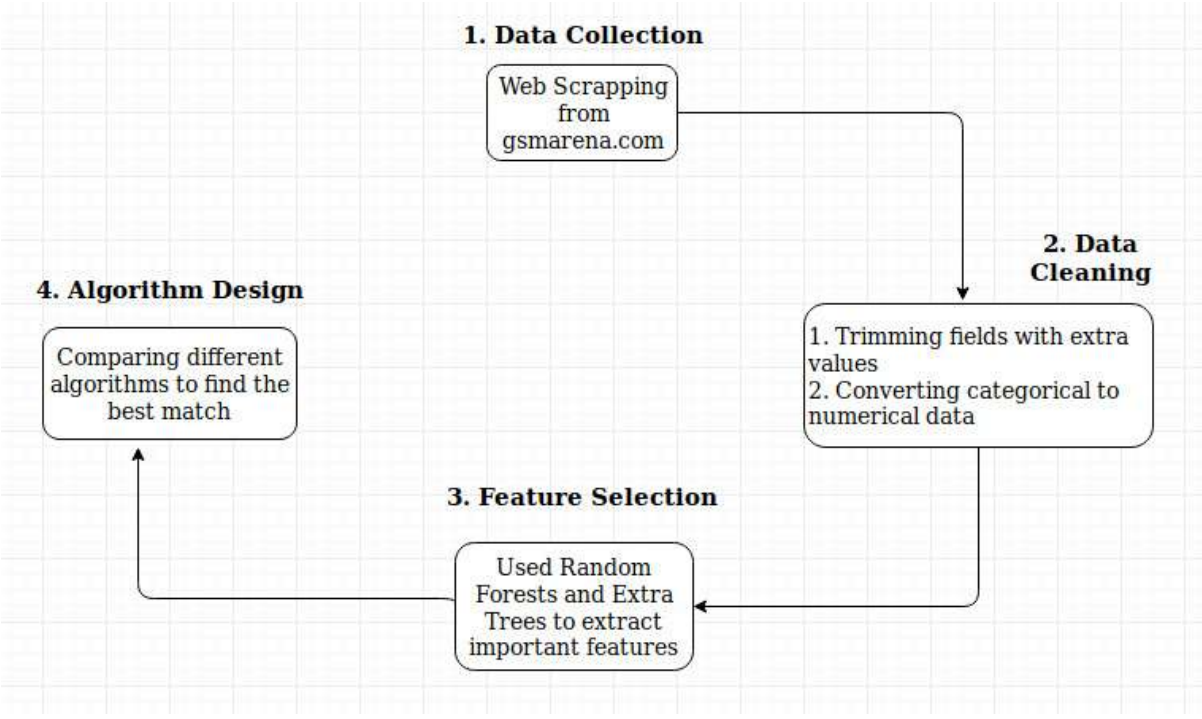
EXPERIMENT 3. (Possible extension of our project)

Compute the features of upcoming phones.

In this experiment, we tried implementing a framework that would predict the features in upcoming smartphones of a particular series. The idea was to try and implement RNN. But we did not get feasible results probably because of small size of dataset.

In further extension we can try collecting dataset of large number of smartphones and apply similar techniques and improve this experiment.

Brief Flow of Project



Task Assignment

1. Dataset Collection

- Tarpit Sahu : 75%
- Anurag Chaturvedi : 25%

2. Dataset Cleaning and Visualization

- Anurag Chaturvedi : 80%
- Tarpit Sahu : 20%

3. Feature Selection and Analysis

- Neelesh Bhakt : 80%
- Shivam Singh : 20%

4. Algorithm Design

- Shivam Singh : 80%
- Neelesh Bhakt : 20%

References

- **www.gsmarena.com (For Dataset Collection)**
- **www.scikit-learn.org (For Algorithm Design)**
- **https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e (Feature Selection Techniques)**