CS 583 — Deep Learning December 9, 2024

Mobile Price Classification

Lecturer: Zhaozhuo Xu By: Tarun Sai,Shiva Raj,Thenapalli Praveen,Manoj Avineni

1 Introduction

The mobile phone market is highly diverse, with a wide range of devices catering to different price segments. Understanding how various features of a mobile phone influence its price can be highly beneficial for manufacturers, retailers, and consumers. Mobile Price Classification involves predicting the price category of a mobile phone based on its specifications, such as battery life, processor speed, RAM, camera quality, and more.

In this project, we aim to classify mobile phones into different price ranges based on their specifications using machine learning techniques. The increasing variety of mobile phones in the market makes it essential to understand the correlation between a phone's features and its price. By leveraging data such as battery life, processor speed, RAM, screen resolution, and more, we can predict the price category of a mobile phone, providing valuable insights for both consumers and manufacturers.

2 Data Collection and Processing

2.1 Data Source

The dataset we used to perform data exploration and modeling of heart disease in patients based on various features, for this project, is available on Kaggle website called Mobile Price Classification which is freely available and can be obtained from Kaggle website using the link: https://www.kaggle.com/code/nimapourmoradi/mobile-price-classification-nn-97-7/input?select=test.csv. The Mobile Price Classification project uses two key datasets: a training dataset and a testing dataset. These datasets contain various mobile phone specifications and their corresponding price categories: Training Dataset (train.csv) and Testing Dataset (test.csv)

- 'battery power' (Battery power in mAh)
- 'Bluetooth' (Bluetooth support (0 for no, 1 for yes))
- 'clock speed' (Speed of the mobile processor in GHz.)
- 'sim' (Support for dual SIM (0 for no, 1 for yes).)
- 'fc' (Front camera megapixels)
- 'four g' (4G support (0 for no, 1 for yes))
- 'int memory' (Internal memory in GB)
- 'mobile wt' (Weight of the mobile phone in gram)

- 'n cores' (Number of processor cores)
- 'px height and px width': (Pixel resolution of the screen)
- 'ram': (RAM in MB.)
- 'sc h and sc w': Screen height and width in cm
- 'talk time': Time the phone can talk on a single charge
- 'three g': 3G support (0 for no, 1 for yes)
- 'touch screen': Touchscreen support (0 for no, 1 for yes)
- 'Wifi': WiFi support (0 for no, 1 for yes
- 'price range': The target variable representing the price category (0 for low cost, 1 for medium, 2 for high, 3 for very high)

2.2 Data Processing

Description: The dataset consists of 21 features related to mobile phone specifications, which include technical details such as battery capacity, processing speed, RAM, screen resolution, and network connectivity. The target variable, price range, represents different price categories of mobile phones, which this project aims to predict. The dataset includes a mix of numerical and binary features, making it suitable for classification tasks

- Handling Missing Data:

  Based on the initial inspection, there are no missing values in the provided datasets (train.csv and test.csv). However, thorough exploratory data analysis (EDA) is recommended to ensure that no data is missing or improperly formatted. Missing data, if found, could be handled using imputation techniques like mean/median imputation for numerical features or mode imputation for categorical/binary features. In this project, since no missing data is initially observed, no imputation may be necessary unless further exploration reveals hidden issues..

- Data Types:The features in the dataset include both binary and numerical variables. The is a classification of some key features based on their types

  Binary Features: These features can take values of 0 or 1, representing "no" or "yes" for certain functionalitie.

  Numerical Features: Continuous numerical values representing different hardware aspects.

- Data Splitting:

  Training and Test Split:

  Since two datasets were provided (train.csv and test.csv), they appear to be pre-split for model training and evaluation. However, within the training dataset, further splitting may be required to create validation data that can be used during the training process.

  Training Set: The model will be trained on the data from train.csv, which includes both the features and the price range target variable.

Validation Set: To prevent overfitting, part of the training data may be set aside for validation purposes (e.g., an 80/20 split). This validation data helps to fine-tune the model and monitor its performance on unseen data before making predictions on the test set.

Test Set: The test.csv file is used to evaluate the final model. Since it does not contain the target variable (price range), the model's predictions on this dataset will be compared to the true values (not provided in the file) to assess its accuracy and generalization capabilities..

- Data Normalization:

  Feature Scaling: Neural networks are sensitive to the scale of the input features. Thus, it is necessary to scale the numerical features such as battery power, ram, px height, px width, etc., to ensure all features are on a similar scale.

- Categorical/Binary Encoding:

  Binary Features: Since the binary features (like dual sim, four g, etc.) are already represented as 0 and 1, there is no need for additional encoding.

  Target Variable Encoding: The target variable, price range, is a categorical variable with four classes (0, 1, 2, 3). In this project, it may be encoded as a one-hot vector (for multi-class classification), but for most classification algorithms, including neural networks, categorical encoding may not be necessary.

Feature Engineering

- Feature engineering may involve transforming or creating new features from the existing ones.

  Combining px height and px width to calculate screen area (height $\times$ width).

  Creating interaction terms between features like ram and clock speed to capture the joint influence on phone performance.

- The data encoding techniques used in this project for converting categorical variables into numerical values are label encoding and one hot encoding so that our machine learning model could effectively interpret and learn from categorical data.

- Effective feature engineering can improve model accuracy by giving it more meaningful inputs.

3 Model Development

3.1 Machine Learning Model Considered

Using Python's Scikit Learn model selection library, the data has been split into two sets, including training sets and testing sets with the split percentage of 80% for training data and 20% for testing data.

In this project, we evaluated several machine learning and deep learning models to predict the price range of mobile phones using their specifications. The selected models were chosen for their ability to capture complex relationships between input features and effectively classify the data into price categories.

- Feedforward Neural Network (FNN): The FNN model serves as a multi-layer perceptron architecture where the data flows in one direction from input to output through hidden layers. It captures non-linear relationships between mobile phone features and their price range. Activation functions like ReLU were used for hidden layers, and softmax for the output layer to classify into four price categories.

- Gradient Boosting: Gradient Boosting is an ensemble method that builds models sequentially, with each model correcting the errors of its predecessor. It is effective for structured data, leveraging weak learners (typically decision trees) to achieve high predictive accuracy.

- Support Vector Classifier (SVC): SVC is a margin-based classification model that separates classes using hyperplanes. The kernel trick allows SVC to capture non-linear relationships in the data, making it well-suited for structured feature spaces.

- Random Forest Classifier: Random Forest is an ensemble of decision trees where each tree is trained on a random subset of data. It reduces overfitting and provides insights into feature importance.

3.2 Chosen Model Architecture

After evaluating the performance of multiple models, the Feedforward Neural Network (FNN) was selected as the final architecture. FNN outperformed other models in accuracy and generalization on unseen data.

3.3 Key Components of the FNN Model Architecture

Input Layer: The input layer accepts numerical and binary features such as RAM, battery power, pixel resolution, clock speed, and more. Features were scaled using Min-Max Normalization to ensure optimal model convergence.

Hidden Layers: Two to three fully connected (dense) layers were used with ReLU (Rectified Linear Unit) activation functions. The hidden layers capture non-linear patterns and interactions between features.

Output Layer: The final output layer uses a softmax activation function to predict the probability of each price category (0, 1, 2, 3). The category with the highest probability is selected as the predicted class.

3.4 Final Model Selection

The FNN model demonstrated the highest accuracy and generalization capability compared to other models. The reasons for selecting FNN as the final model are:

Performance: FNN achieved the best trade-off between accuracy, precision, recall, and F1-score. Flexibility: The model effectively captured non-linear relationships in the data and performed well with both numerical and binary features. Generalization: FNN was able to

avoid overfitting through regularization techniques like dropout and performed consistently on training, validation, and test datasets.

4 Results Analysis

Feedforward Neural Network (FNN): The FNN model achieved the highest accuracy of 92.5Its ability to capture complex, non-linear patterns in the data contributed to its superior performance.

Random Forest: The Random Forest model performed well with an accuracy of 90.2Feature importance analysis showed that RAM and Battery Power were the most influential features.

Gradient Boosting and SVC: Gradient Boosting achieved a competitive accuracy of 89.5SVC, though slightly lower at 88.0This analysis clearly demonstrates that FNN is the most effective model for predicting mobile phone price categories. Its robust architecture and non-linear learning capabilities make it ideal for handling the feature set in this dataset. .

5 Conclusions

The project aimed to predict the price range of mobile phones based on their specifications using various machine learning and deep learning models. After exploring models such as Gradient Boosting, Support Vector Classifier (SVC), Random Forest, and Feedforward Neural Network (FNN), the FNN model demonstrated superior performance across all evaluation metrics, including accuracy, precision, recall, and F1-score.

The FNN model achieved an impressive accuracy of 92.5

Key features such as RAM, Battery Power, and Pixel Resolution were identified as the most influential predictors of the mobile price range. These insights align with real-world expectations, where performance and display quality play a significant role in determining mobile phone pricing.

In summary, the Feedforward Neural Network (FNN) proved to be the optimal solution for this classification task, achieving reliable and accurate predictions. This project demonstrates the power of deep learning in solving structured data problems and highlights its potential for applications in price prediction and decision-making systems.

References

1. D. S. Rana, S. A. Dhondiyal, S. Singh, S. Kukreti and A. Dhyani, "Predicting Mobile Prices with Machine Learning Techniques," 2024 International Conference on Computational Intelligence and Computing Applications (ICCICA), Samalkha, India, 2024, pp. 248-252, doi: 10.1109/ICCICA60014.2024.10585222. keywords: Support vector machines;Machine learning algorithms;Costs;Machine learning;Pricing;Predictive models;Prediction algorithms;Machine learning;SVM Method;Decision Tree;Phone Price Prediction,

2. A. K. Bishnoi and S. K. Mandal, "Performance Analysis of Decision Tree Models and M5P Models for Mobile Phone Price Prediction," 2023 International Conference on Advances in

Computation, Communication and Information Technology (ICAICCIT), Faridabad, India, 2023, pp. 1298-1302, doi: 10.1109/ICAICCIT60255.2023.10466198. keywords: Analytical models;Costs;Computational modeling;Machine learning;Predictive models;Mobile handsets;Performance analysis;Predictive analytics;regression;decision tree;M5P,

3. N. Hu, "Classification of Mobile Phone Price Dataset Using Machine Learning Algorithms," 2022 3rd International Conference on Pattern Recognition and Machine Learning (PRML), Chengdu, China, 2022, pp. 438-443, doi: 10.1109/PRML56267.2022.9882236. keywords: Support vector machines;Machine learning algorithms;Random access memory;Machine learning;Feature extraction;Mobile handsets;Batteries;computer science;machine learning;classification;price prediction

4. M. Kumar, U. Pilania and C. Varshney, "Predicting Mobile Phone Prices with Machine Learning," 2023 3rd International Conference on Advancement in Electronics Communication Engineering. keywords: Industries;Machine learning algorithms;Pricing;Prediction algorithms;Mobile handsets;Classification algorithms;Random forests;Prediction;Decision Tree;Random Forest;K-Nearest Neighbors;Accuracy,

5. . Çetın and Y. Koç, "Mobile Phone Price Class Prediction Using Different Classification Algorithms with Feature Selection and Parameter Optimization," 2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Ankara, Turkey, 2021, pp. 483-487, doi: 10.1109/ISMSIT52890.2021.9604550. keywords: Machine learning algorithms;Static VAr compensators;Support vector machine classification;Feature extraction;Prediction algorithms;Mobile handsets;Classification algorithms;Regression;Classification;Feature Selection;Parameter Optimization;ANOVA;Mutual Information;Random Forest Classifier;Logistic Regression Classifier;Decision Tree Classifier;Linear Discriminant Analysis;KNN Classifier;Support Vector Machine,

6. J. Sun and D. Wu, "Contextual-Feature-Based Budget-Limited Online Pricing for Heterogeneous Sensing Tasks," in IEEE Internet of Things Journal, vol. 11, no. 9, pp. 15783-15791, 1 May1, 2024, doi: 10.1109/JIOT.2023.3348506. keywords: Task analysis;Pricing;Sensors;Costs;Cost accounting;Recruitment;Cost function;Budget-limited pricing;contextual feature;heterogeneous sensing tasks;Lipschitz cost,

7. Y. Zhang, Q. Ding and C. Liu, "An Enhanced XGBoost Algorithm for Mobile Price Classification," 2023 IEEE 6th International Conference on Big Data and Artificial Intelligence (BDAI), Jiaxing, China, 2023, pp. 154-159, doi: 10.1109/BDAI59165.2023.10256847. keywords: Radio frequency;Dimensionality reduction;Filtering;Mobile handsets;Hardware;Decision trees;Optimization;XGBoost;Dung beetle optimizer (DBO);Parameter optimization;Classification,