**SAVITRIBAI PHULE PUNE UNIVERSITY**



**A**

**PROJECT REPORT**

**ON**

**“PROJECT TITLE”**

***A project work submitted to the Department of Computer Engineering, SITS , Narhe , Pune. In the fulfillment of the requirements for***

**SOFTWARE DEVELOPMENT LAB**

**Third Year (Computer Engineering)**

**By**

Under the guidance of

**DEPARTMENT OF**

**COMPUTER ENGINEERING**

**SINHGAD INSTITUTE OF TECHNOLOGY & SCIENCE, NARHE, PUNE**

**(2020 – 2021)**

**SINHGAD TECHNICAL EDUCATION SOCIETY’S**

**SINHGAD INSTITUTE OF TECHNOLOGY AND SCIENCE**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Name of the Chapter** | **Page No.** |
|  | **Introduction** |  |
|  | **Software Development Life cycle** |  |
|  | **Requirements (Hardware, Software)** |  |
|  | **Design and Modeling** |  |
|  | **Implementation** |  |
|  | **Testing** |  |
|  | **Maintenance** |  |
|  | **Conclusion** |  |

**1. INTRODUCTION**

* Many people get scammed while buying a used car over the quality of the car they are buying.
* The quality evaluation system helps one in confirming whether the car they are buying is worth it or not.

**1.1 Evolution:**

This project is helpful in preventing them getting scammed by the dealers. So, we will build a system using Python, sklearn which will alert the driver when he feels sleepy.

**1.2 Proposed System**:

This system will detect the car, run it through its system and give us an analysis on whether the car acceptable or not. This system will be very useful to prevent scams by alerting the buyer on whether it’s a good deal or not and categorise the quality in unacceptable, acceptable, good or very-good.

**1.3 Software and Hardware Requirements:**

The Software and Hardware Requirements are stated as below:

● **Technical Hardware Requirement**

|  |  |
| --- | --- |
| **Name** | **Details** |
| Processor | Intel Pentium 4 CPU |
| RAM | 512 MB |

**Table 1.3.1 Hardware Requirements**

Hardware Requirements are the necessary tools that must be available for project to be done properly. This System is built on Intel Pentium 4 CPU, having clock speed of 3.0GHz, with RAM size 512MB, display is of 15-inch color monitor, and internet keyboard.

* **Software Requirement**

|  |  |
| --- | --- |
| **Name** | **Details** |
| Operating System | Windows OS/Linux |
| Platform | Pycharm |
| Language | Python |

**Table 1.3.2 Software Requirements**

The documentation of this system is done using MS-Office.

**2. Project Description**

In this Python project, we will be using Sklearn for gathering the images from the input and feed them into a Deep Learning model which will classify whether the car’s quality is ‘Unacceptable’, ‘Acceptable’, ‘Good’, ‘Very good’. The approach we will be using for this Python project is as follows

Step 1 – Take image as input from the user.

Step 2 – Detect the car in the image and analyze the car.

Step 3 – Detect the quality of the car.

Step 4 – Evaluate the quality of the car.

Step 5 – Give the graphical representation of the analysis.

* **The Dataset**

The dataset used for this model is created by us. To create the dataset, we wrote a script that captures the image of the car from the input and stores in our local disk. We separated them into their respective labels ‘Open’ or ‘Closed’. The data comprises around 1750 quality checks on cars.

Now, you can use this model to check the quality of a used car.

* **Prerequisites**

The requirement for this Python project is a GUI through which we will insert images. You need to have Python (3.6 version recommended) installed on your system, then using pip, you can install the necessary packages.

Pandas – pip install pandas-python (data manipulation).

MatplotLib - pip install matplotlib(To plot a graph)

sklearn - pip install sklearn(for Machine learning)

seaborn - pip install seaborn (for data visualization)

numpy -pip install numpy (for working with arrays)

**3 Requirements (Hardware, Software):**

Building a machine learning / deep learning workstation can be difficult and intimidating. There are so many choices out there. Would you go for NVidia developer box and spend $15,000? or could you build something better in a more cost-effective manner. Which hardware is right for your requirements? How much RAM do you need? The questions are endless and there might be no right or wrong answers.

* **Hardware Requirements:**
* Processor – Intel I3 2 core processor, 2.2 GHz with Turbo-boost up to 3.1 GHz.
* Motherboard – Intel DH61HO.
* RAM – 4 GB DDR4 2133 MHz
* 2 TB Hard Disk (7200 RPM) + 512 GB SSD.

**4. Technical Documentation**

1. **Pandas**

Pandas is a high-level data manipulation tool developed by Wes McKinney. It is built on the Numpy package and its key data structure is called the DataFrame. DataFrames allow you to store and manipulate tabular data in rows of observations and columns of variables.

1. **Sklearn**

**Scikit-learn** (formerly **scikits.learn** and also known as **sklearn**) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, *k*-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

Scikit-learn is largely written in Python, and uses numpy extensively for high-performance linear algebra and array operations. Furthermore, some core algorithms are written in Cython to improve performance. Support vector machines are implemented by a Cython wrapper around LIBSVM; logistic regression and linear support vector machines by a similar wrapper around LIBLINEAR. In such cases, extending these methods with Python may not be possible.

Scikit-learn integrates well with many other Python libraries, such as matplotlib and plotly for plotting, numpy for array vectorization, pandas dataframes, scipy, and many more.

1. **Matplotlib**

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of Matplotlib.

1. **Numpy**

NumPy is a python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely. NumPy stands for Numerical Python. NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently. This behavior is called locality of reference in computer science. This is the main reason why NumPy is faster than lists. Also, it is optimized to work with latest CPU architectures.

1. **Seaborn**

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

**5. IMPLEMENTATION**

**Let’s now understand how our algorithm works step by step.**

**Step 1 – Take input from the user.**

With the help of a GUI, we will take the input. We loaded the trained model in the GUI to help with analyzation

**Step 2 – Detect the car in the image.**

The car detected and then goes in the system for analyzing.

**Step 3 – Detect the quality of the car.**

After detection of the car, it is then analyzed and the data is collected with the help of sklearn.

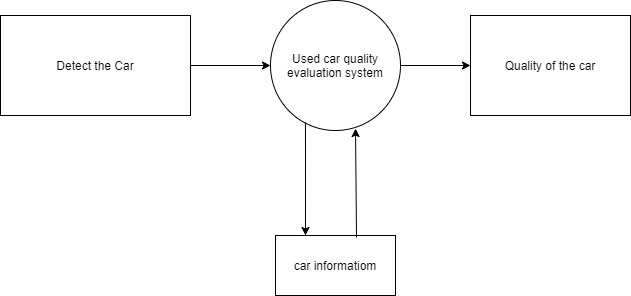
**Step 4 – Evaluate the quality of the car.**

After the data is collected, it is compared with the train model and the result is given as ‘unacceptable’ or ‘acceptable’ or ‘good’ or ‘very-good’ and then a graph of accuracy gets plotted.

**Step 5 – Give the graphical representation of the analysis.**

The result is the displayed in the form of a confusion matrix.

**6. Data Flow Diagram**

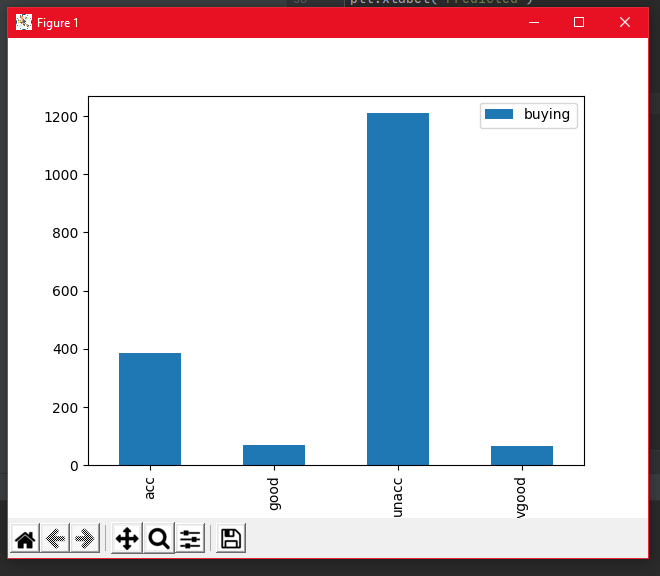
****

**7. Result**

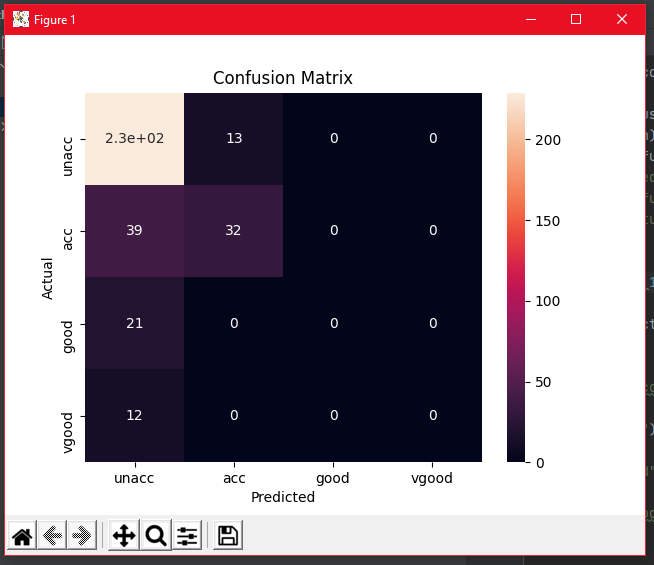
Let’s start our project and see the working of our project. To start the project, you need to open a command prompt, go to the directory where our main file “det.py” exists. Run the script with this command.

**Output Screenshot:**

1. Accuracy graph after complete evaluation of the car



**2. Result in the form of confusion matrix**

****

**8. CONCLUSION**

* In this Python project, we have built a system in order to prevent scams by sending analysis on quality of the cars and can be implemented in numerous ways. We used sklearn to detect the quality of the car and matplotlib, pandas and seaborn to represent the analysis.

.