

GERENCIAMENTO DA PRODUÇÃO DE UMA FÁBRICA DE CELULARES

Universidade Federal do Rio de Janeiro

Departamento de Engenharia Eletrônica e de Computação

EEL670 – Linguagens de Programação

Professor: Miguel Campista

Autores: Felipe Huan Kuang

Daniel Matheus Peclat

1. Introdução

Este projeto foi realizado no intuito de simular um controle de estoque de uma fábrica de celulares. O trabalho foi realizado na linguagem C++ em conjunto com funções externas escritas em Python e o conhecimento em aprendizado de máquina.

2. Objetivo

As fábricas realizam produções em massa de diversos produtos encontrados no mercado. Portanto, é comum que haja falhas no controle de qualidade, ou seja, falhas no processo de produção podem ocorrer, gerando produtos defeituosos.

Produtos defeituosos podem passar despercebidos facilmente, o que pode acarretar em futuros problemas para a empresa. Portanto, faz-se necessário um programa de gerenciamento e controle da produção para evitar que problemas, como mencionado anteriormente, passem despercebidos e assim assegurando a qualidade e credibilidade na linha de produção da empresa.

3. Programa

O programa foi implementado em linguagem C++ e Python. O primeiro é responsável pelo programa gerenciador e o segundo pelos scripts relacionados ao teste de acurácia da produção e o classificador de qualidade do produto.

O programa foi desenvolvido a partir da versão Python3.8, sendo obrigatória a instalação das bibliotecas <python3.8/Python.h> por meio do comando pip3. Além disso, necessita-se adicionar na variável de ambiente: `export PYTHONPATH="$PWD"`.

O programa consiste da estruturação do menu, que é responsável pela interação com o usuário, e criação da classe e struct responsáveis pelo armazenamento dos celulares no estoque. Para inicializar o programa, é retirado de um arquivo .txt, de nome padrão "codigoCelulares.txt", os nomes dos arquivos .txt correspondentes a cada código de celular.

4. Casos de uso

Ao iniciar o programa, o usuário terá acesso ao menu com as seguintes opções:

- 1 – Exibir uma lista com os modelos existentes de celular;
- 2 – Inserir um modelo novo de celular;
- 3 – Remover um modelo existente de celular;
- 4 – Verificar se um celular é defeituoso ou se ele está em bom estado;
- 5 – Verificar acurácia do programa de classificação do produto;
- 0 – Finalização do programa.

Comando 1 – Exibir uma lista com os modelos existentes de celular:

Este comando exibe uma lista completa com os modelos de celulares existente e a descrição de cada um deles.

```
Esses sao os modelos de celular presentes no estoque atual:  
  
Modelo: Celular Apple  
Descrição:  
Tela rachada  
  
Modelo: Celular Samsung  
Descrição:  
Otimo estado  
  
Modelo: Celular Motorola  
Descrição:  
Celular com defeito
```

Figura 1: Lista com modelos de celular cadastrados

Comando 2 – Inserir um novo modelo de celular:

Neste comando, será pedido ao usuário que insira o nome do novo modelo de celular a ser adicionado no sistema e a descrição do mesmo.

```
Modelo: Celular Xiaomi  
Descrição do celular (estado/qualidade do produto): <Digite em uma linha "FIM" para terminar o processo de escrita>  
Tela Rachada  
FIM  
Código do celular: celular4.txt
```

Figura 2: Inclusão de um novo modelo de celular

O usuário poderá escrever uma descrição do produto, e quando finalizado, devera digitar “FIM” para que ele possa definir o código de busca do modelo no sistema.

Comando 3 – Remover um celular da lista:

Neste comando, será mostrado uma lista com os códigos de todos os celulares cadastrados no sistema.

```
Digite o codigo do Celular que deseja apagar. Os modelos disponiveis estao listados abaixo:  
celular1.txt  
celular2.txt  
celular3.txt  
celular4.txt  
celular4.txt
```

Figura 3: Listagem com celulares cadastrados

O usuário deverá digitar o código do produto o qual deseja remover, e o produto será removido do sistema.

Comando 4 – Verificar o estado do produto

Essa opção é responsável pela classificação dos produtos por meio de um algoritmo classificador. O programa exibirá todos os códigos disponíveis para teste do usuário e este deverá entrar com o nome do arquivo desejado. Em seguida, o classificador exibe na tela o resultado da classificação se o código é de um produto defeituoso ou de um celular ok (em bom estado). Após a exibição, o usuário retorna ao menu principal.

```
Digite o codigo do celular desejado. Os modelos disponiveis estao listados abaixo:  
(A verificacao sera feita por um algoritmo externo)  
  
celular1.txt  
celular2.txt  
celular3.txt  
celular1.txt  
Resultado: ['defeituoso']
```

Figura 4: Funcionamento do Classificador

Comando 5 – Verificar a acurácia do programa de classificação

A quinta opção é responsável pelo teste do grau de acurácia do classificador. O resultado é exibido na tela em número decimal (de 0 a 1). Após a exibição, o usuário retorna ao menu principal.

```
Modelo SVM - Acuracia: 0.81818181818182
```

Figura 5: Acurácia do classificador

Comando 0 – Saída do programa

Caso o usuário entre com '0' no menu principal, é acionada a opção, que é responsável pelo encerramento do programa. Ao pressionar '0' e então o programa exibirá uma mensagem em processo de encerramento e executará todos os destrutores default.

5. Conclusão

Por meio dos testes que realizamos, pudemos verificar o funcionamento do nosso programa como planejado inicialmente. Durante o procedimento de elaboração do projeto, a pesquisa de bibliotecas e funções foi necessária para a sua conclusão. O aprendizado da integração entre C++ e Python via API foi fundamental para a correta execução dos scripts a partir do C++, assim como para a passagem de parâmetro entre as linguagens.

6. Referências

[1] activestate.com, “How to install and use pip3”, <https://www.activestate.com/resources/quick-reads/how-to-install-and-use-pip3/>

[2] scikit-learn.org, “CountVectorizer”, https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

[3] scikit-learn.org, “Support Vector Machines”, <https://scikit-learn.org/stable/modules/svm.html>

[4] docs.python.org, “Extending Python with C or C++”, <https://docs.python.org/3/extending/extending.html>