Multilabel Classification Project for Predicting Shipment Modes
Project Overview

Overview

The transport industry is a critical component of the global economy. The efficient movement of goods is necessary to ensure that businesses can operate effectively and customers can receive their products on time. However, determining the appropriate mode of transport for each shipment can be challenging. It requires considering various factors such as the type of product being transported, the distance, and the destination. Choosing the appropriate mode of transport for each shipment can significantly affect the delivery time, cost, and safety of the goods being transported.

For example, air transport is generally faster but more expensive than other modes, while sea transport is slower but more cost-effective for large shipments. The wrong choice of transport mode can result in delays, damage to the goods, or increased costs for the business. By accurately predicting the appropriate mode of transport for each shipment, businesses can optimize their logistics operations, reduce costs, and improve customer satisfaction.

To build the machine learning model, we first explore the dataset using BigQuery. We will implement four different approaches to multilabel classification:

- **Naive independent models** - We will build independent models for each label and combine the predictions to determine the appropriate mode of transport.

- **Classifier chains** - We will use a chain of classifiers, where the output of one classifier is fed into the next classifier to predict the labels.

- **Natively multilabel models** - We will use models designed to handle multilabel classification tasks, such as Extra Trees and Neural Networks.

- **Multilabel to multiclass approach** - We will convert the multilabel problem into a multiclass problem by combining different combinations of labels and training the model to predict these combinations. After prediction, we will separate the combinations back into individual labels.

Aim

This project aims to develop a machine-learning model that can predict the appropriate mode of transport for each shipment. We will explore and compare different approaches to multilabel classification for this task.

Data Description

The transport dataset contains 2000 unique products with the following columns:

- Product_Id: Unique identifier for each product.
- Net_Weight: The weight of the product.
- Size: The size of the product, with options A, B, C, D, and E.
- Value: The value of the product.
- Storage: Binary value indicating whether the product requires special storage conditions.
- Packaging_Cost: The cost of packaging the product.
- Expiry_Period: The expiry period of the product.
- Length: The length of the product.
- Height: The height of the product.
- Width: The width of the product.
- Volume: The volume of the product.
- Perishable_Index: Value indicating the perishability of the product.
- Flammability_Index: Value indicating the flammability of the product.
- F145, F7987, F992: anonymous numerical columns.
- Air, Road, Rail, Sea: Binary values indicate the appropriate transport mode for the product.

Tech Stack

➔ Language: Python

➔ Libraries: pandas, numpy, matplotlib, scikit-learn, tensorflow, Google BigQuery.

Approach

- Exploratory Data Analysis (EDA):
    - Understand the features
    - Check the data summary
    - Check for missing or invalid values

- Preprocessing:
    - Encoding the categorical features

- Split the dataset into training and testing sets
- Create cross-validation sets
- Multilabel Classification:
    - Approach 0 - Naive Independent Models:
        - Train separate binary classifiers for each target label-lightgbm
        - Predict the label
        - Evaluate model performance using the f1 score
    - Approach 1 - Classifier Chains:
        - Train a binary classifier for each target label
        - Chain the classifiers together to consider the dependencies between labels
        - Predict the label
        - Evaluate model performance using the f1 score
    - Approach 2 - Natively Multilabel Models:
        - Train models that can natively handle multiple labels
        - Use models such as Extra Trees and Neural Networks
        - Evaluate model performance using the f1 score
    - Approach 3 - Multilabel to Multiclass Approach:
        - Combine different combinations of labels into a single target label
        - Train a lightgbm classifier on the combined labels
        - Evaluate model performance using f1 score, precision, and recall

Modular code overview:

```
data
|_transport_shipment_data.csv

lib
|_multilabel_classification.ipynb

ml_pipeline
|_utils.py
|_processing.py
|_training.py

engine.py

requirements.txt

readme.md
```

Once you unzip the modular_code.zip file, you can find the following folders.

1. data
2. lib
3. ml_pipeline
4. engine.py
5. requirements.txt
6. readme.md

1. The lib folder is a reference folder and contains the original ipython notebook as in the lectures.

2. The ml_pipeline folder contains all the functions put into different python files, which are appropriately named. The engine.py script then calls these python functions to run the steps in one go to train the model and print the results.

3. The requirements.txt file has all the required libraries with respective versions. Kindly install the file using the command **pip install -r requirements.txt**

4. **All the instructions for running the code are present in readme.md file**

Project Takeaways

1. Understanding the importance of predicting the appropriate mode of transport for each shipment in the transport industry.
2. Exploring a multilabel classification problem statement.
3. Understanding the four different approaches to multilabel classification.
4. Querying and exploring the dataset using BigQuery.
5. Preprocessing and cleaning the dataset using GCP Big Query
6. Feature selection and engineering using domain knowledge.
7. Implementing naive independent models for each label and evaluating the results.
8. Implementing classifier chains for multilabel classification and evaluating the results.
9. Implementing natively multilabel models, such as Extra Trees and Neural Networks, and evaluating the results.
10. Implementing the multilabel to multiclass approach and evaluating the results.
11. Comparing and contrasting the results of each approach.
12. Understanding the trade-offs and limitations of each approach.