# Abstract

Maritime shipping has now become an important part of worldwide trade as a result of globalization. As a result, the reliability of cargo vessel shipping company has become an important element in the shipping business, as inconsistencies may cause additional delays farther down the supply chain, raising the entire cost of shipping goods or may destroy the shipping goods. In this project, we are using different built-in deep learning models for the prediction of best suitable service provider base on the different features of Cargo container. Here, we also proposed a new deep learning model to more accurately predict the suitable service provider.

# Dataset Description

## Introduction:

The dataset of shipping cargos with shipment details was downloaded from Kaggle. The dataset contained the several features related to the cargos with the name of the service provider company. The downloaded dataset was used in this project for the prediction of best suitable service provider company on the basis of cargo features.

The original Dataset contains the following number of rows and Columns:

|  |  |  |
| --- | --- | --- |
| **Voyage Dataset** | **Number of Rows in Dataset:** | 21590 |
| **Number of Variables in Dataset:** | 15 |

# Dataset – Summary of Attributes

The both datasets were saved in excel file and the variable detail of both datasets is available in the following table. The table described the variable name, description of variable, and type of variable.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Variable Name | Description | Type |
| 1 | Vessel Name | Name of the Vessel | Categorical |
| 2 | Vessel Type | Type of the Vessel (On the basis of carrying goods) | Categorical |
| 3 | Vessel Capacity | Maximum Size or capacity of vessel of carrying goods | Numeric |
| 4 | Load Date | The date on which vessel was loaded with goods | Date |
| 5 | Category of Trade | Category of goods that was trade. | Categorical |
| 6 | Cargo Description | Brief description about the cargo. | Text |
| 7 | Volume/ Amount | Size or volume of the goods in the cargo. | Numeric |
| 8 | Volume Type | Unit of measuring the volume of the goods in Cargo | Categorical |
| 9 | Load Port | Name of the port from the cargo was loaded. | Categorical |
| 10 | Discharge Port | Destination Port of cargo | Categorical |
| 11 | Dangerous Goods | Flag for indicating that the cargo is containing the Dangerous goods or not? | Boolean |
| 12 | Organization | Name of the Organization that provide or trade the cargo from load port to destination port. | Categorical |
| 13 | Application Number | Identity number provided by organization | Numeric |
| 14 | License Number | License number of the organization. | Numeric |
| 15 | Voyage Number | Identity Number of the Cargo/Vessel | Numeric |

# Data Cleaning

## Remove Identical Features

As the dataset have the total 15 features, but there are several features that are base on identity and does not correlate with the other attributes. In other words, these attributes are unique identity n umber and did not affect the any other features. So, all the features that have static effect were removed from the dataset. The status of all the features after this procedure is presented in following table.

|  |  |  |  |
| --- | --- | --- | --- |
| Variable Name | Status | Variable Name | Status |
| Vessel Name | Removed | Load Port | Keep |
| Vessel Type | Removed | Discharge Port | Keep |
| Vessel Capacity | Removed | Dangerous Goods | Keep |
| Load Date | Removed | Organization | Keep |
| Category of Trade | Keep | Application Number | Removed |
| Cargo Description | Removed | License Number | Removed |
| Volume/ Amount | Keep | Voyage Number | Removed |
| Volume Type | Keep |  |  |

## Attribute data types and Statistics

The Voyage report datasets have the total 15 attributes. The datatype of each attribute with the elementary statistics is following:

|  |  |  |  |
| --- | --- | --- | --- |
| Voyage Dataset | | | |
|  | **Variable Name** | **Type** | Statistics |
| 1 | Vessel Name | Categorical |  |
| 2 | Vessel Type | Categorical |  |
| 3 | Vessel Capacity | Numeric |  |
| 4 | Load Date | Date |  |
| 5 | Category of Trade | Categorical |  |
| 6 | Cargo Description | Text |  |
| 7 | Volume/ Amount | Numeric |  |
| 8 | Volume Type | Categorical |  |
| 9 | Load Port | Categorical |  |
| 10 | Discharge Port | Categorical |  |
| 11 | Dangerous Goods | Boolean |  |
| 12 | Organization | Categorical |  |
| 13 | Application Number | Numeric |  |
| 14 | License Number | Numeric |  |
| 15 | Voyage Number | Numeric |  |

## Target Variable Description

After the cleaning of the dataset, the “Organization” feature was set as target variable datasets. The target variable contains the different number of samples for 81 unique organizations that provide cargo services. The target variable was used for the prediction of best suitable service provider based on the other attributes in dataset. The comprehensive information of features and target variable is available in below table.

|  |  |
| --- | --- |
| Features | Category of Trade, Volume Type, Load Port, Discharge Port, Dangerous Goods |
| Target | Organization |

As the target variable in Voyage dataset is in Categorical form that require the label encoding. Label encoding is a technique of converting string categorical value into numerical value.

## Identification of missing values and remove outliers

Missing value in the tabular data usually contain the NAN, undefined and None values. All the features were pass to the filter to check the NAN values and fill it with mean value of that feature. But the dataset was well structured and clean enough that the filter did not find any missing value.

# Quality of Data

The final set of data contains 7 columns for Voyage dataset. The total numbers of observations (rows) remaining in dataset is 21590.

Columns have been checked for the existence of extreme values (outliers) and actions have been taken  
to remove these observations out of the dataset. This decision has not been taken lightly. While outliers  
may indicate something scientifically interesting; in this scenario – where the objective is to simply  
predict a categorical variable using a classification decision tree; it is believed the presence of outliers  
may hinder the performance of the classifiers.

Furthermore, columns have also been checked for missing values. In particular the removal of missing  
values started by taking into consideration priority columns such as the target variable column  
precipitation type. Since the absence of a value within that column will render this column as impractical  
a decision has been taken to remove these observations out of the dataset.  
Additionally, columns have been checked for content validity by removing observations where incorrect  
data has been identified.

# Train Test Split of Dataset

Furthermore, the original dataset was split into training, testing and validation set with the ratio of 70%, 10% and 20% respectively. For the division of the dataset, the train test split function of python library (scikit-learn) was used that shuffle the dataset and randomly split the samples in each set. This approach was used to split the dataset with balance samples of each class. The training and validation set was usually used during the training of the model while the testing set was used for the testing of the models. The total number of variables and samples in training, testing and validation set is given below:

|  |  |  |
| --- | --- | --- |
|  | **No. of Samples** | **No of Columns** |
| **Training Set** | 15155 | 7 |
| **Testing Set** | 2159 | 7 |
| **Validation Set** | 4275 | 7 |

# Deep Learning Model Implementation

After the cleaning and preprocessing of the datasets, identify the machine learning problem for target variable. As the target variable in Voyage datasets have the Categorical value that indicate that the prediction of target variable made possible by classification models. Hence, different deep learning models for target variable prediction were chosen.

The list of all chosen regression models is below:

* TabNet
* CNN
* Tabular
* DeepTables
* 3 Layer MLP

After the training of the models, all models were tested on test set of the dataset to evaluate the performance of trained models. Below are some evaluation measures that were used to evaluate the performance of the model.



**Accuracy -** Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.

**Precision -** Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

**Recall (Sensitivity) -** Recall is the ratio of correctly predicted positive observations to the all observations in actual class - yes.

**F1 score -** F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as accuracy, but F1 is usually more useful than accuracy, especially if you have an uneven class distribution.

## TabNet

TabNet is a Neural Network model of deep learning for classification of images, text and structural data. In this project, we used the TabNet model for the prediction of best service providers. Next, we initialize the TabNet model for the classification of best service providers. The different values of learning rate, optimizer and batch size were used for the training of the model while rest of the hyper parameters were used with default values. The training and validation sets were used to train and validate the model during training process. The importance of features for the Prediction of service provider using TabNet model is showed below:



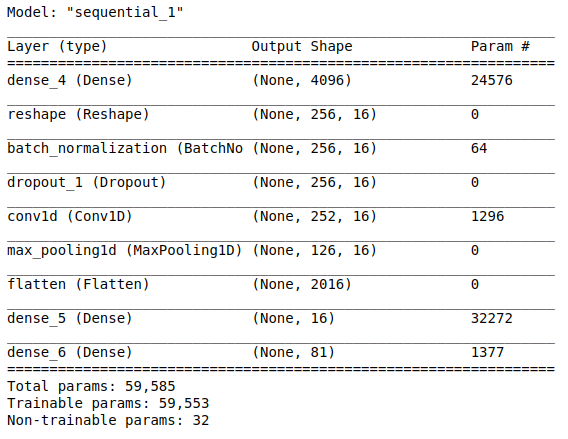
The total six input features of 15155 samples in training set were passed for the training of the model while 15155 labels of training samples were also passed as output variable. For the validation of the model, the features and class target variable of validation set was also passed to the model. TabNet model showed the best result in term of accuracy for optimizer as Adam, learning rate: 0.020 and batch size: 256.

After implementing the combination of different hyper parameters, the TabNet model showed the best 0.7347 accuracy for validation data w. The accuracy and loss plot of TabNet model is presented below:

|  |  |
| --- | --- |
|  |  |

## CNN

CNN is a convolutional neural network base deep learning model for the classification of images and structural or tabular data. In this project, we used the CNN model for the prediction of best service providers. Next, we initialize the CNN model for the classification of best service providers. CNN model was developed by using the different layer including the dense layer, flatten layer, MaxPooling layer, dropout layer and Batch Normalization layer. The architecture of CNN model is also shown in below figure.



The different values of learning rate, optimizer and batch size were used for the training of the model while rest of the hyper parameters were used with default values. The training and validation sets were used to train and validate the model during training process. For the training of the CNN model, the early stopping mechanism was implemented by monitoring the accuracy of the model. The accuracy and loss function graph of training data and validation data is shown below.

|  |  |
| --- | --- |
|  |  |

The above result is the best result produced by the CNN model. CNN model showed the best result in term of accuracy with Adam optimeter, 1000 batch size and 0.01 learning rate. CNN model showed the 0.7284% validation accuracy with early stopping mechanism.

## Tabular

Tabular is the advance form of machine learning model on the idea of XGBoost model by using pyt6orch library. In the proposed project, tabular model was also used for the classification of best service provider. Next, we initialize the CNN model for the classification of best service providers. The different values of learning rate, optimizer and batch size were used for the training of the model while rest of the hyper parameters were used with default values. The training and validation sets were used to train and validate the model during training process.

After the training of the model, tabular model showed the beast result with Adam optimizer, 0.0001 learning rate and 1024 batch size. The best accuracy that was predicted by the tabular model is 0.717 for validation data.

## DeepTables

DeepTables is the advance form decision trees by implementing the deep learning layers. In this project, we used the DeepTables model for the prediction of best service providers. Next, we initialize the DeepTables model for the classification of best service providers. DeepTables model was developed by using the different layer including the dense layer, flatten layer, MaxPooling layer, dropout layer, Batch Normalization layer and embedding layers.

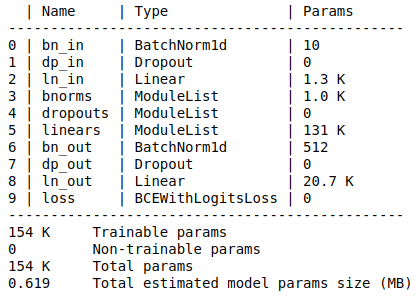
The different values of learning rate, optimizer and batch size were used for the training of the model while rest of the hyper parameters were used with default values. The training and validation sets were used to train and validate the model during training process. For the training of the DeepTables model, the early stopping mechanism was implemented by monitoring the accuracy of the model. The accuracy and loss function graph of training data and validation data is shown below.

|  |  |
| --- | --- |
|  |  |

The above result is the best result produced by the DeepTables model. DeepTables model showed the best result in term of accuracy with Adam optimeter, 1 batch size and 0.01 learning rate. DeepTables model showed the 0.4590% validation accuracy with early stopping mechanism.

## 3 Layer MLP

3 Layer MLP is a neural network based deep learning model that used different dense layer in its architecture. The architecture of the model included the dense layers, dropout layer, normalization layers and liner layers. The architecture of the 3 Layer MLP model is presented below:

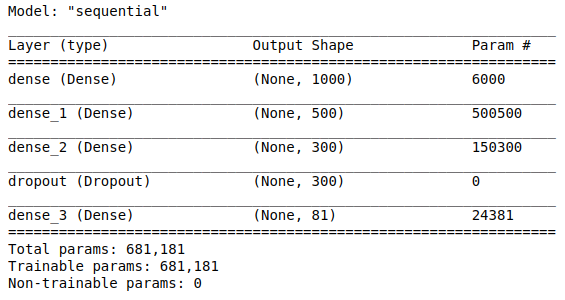


The training process include the training and hyper parameter tunning of the model. The training and validation sets were used to train and validate the model during training process. The different values of learning rate, optimizer and batch size were used for the training of the model while rest of the hyper parameters were used with default values. The early stopping technique was also used with 0.01 patience and accuracy as monitoring matrix to get the optimal result of the model.

The above result is the best result produced by the DeepTables model. DeepTables model showed the best result in term of accuracy with Adam optimeter, 1 batch size and 0.01 learning rate. DeepTables model showed the 0.4170% validation accuracy with early stopping mechanism.

## Customize Model

Lastly, we proposed our own model for the prediction of best service provider. The customized model is a Neural network base deep learning model base on the different layers of the deep learning. The model was developed in TensorFlow by using the dense layers of different sizes. The complete architecture of customized model is shown in below figure.



For our customize model, the training and validation sets were used during the training of the model. The different values of learning rate, optimizer and batch size were used for the training of the model while rest of the hyper parameters were used with default values. The early stopping technique was also used with 0.01 patience and accuracy as monitoring matrix to get the optimal result of the model. The accuracy and loss function graph of training data and validation data is shown below.

|  |  |
| --- | --- |
|  |  |

The above result is the best result produced by the DeepTables model. DeepTables model showed the best result in term of accuracy with Adam optimeter, 1 batch size and 0.01 learning rate. DeepTables model showed the 0.0.7934% validation accuracy with early stopping mechanism.

# Testing

All the models were test against the testing set of the both datasets. The testing set was passes to all the trained model for making predictions. The testing set was used to calculate the score of each evaluation measure for every model by using the evaluation equations (describe above).

**Testing Set –** Testing set was based on the total six attributes like the training set. The Organization variable in testing set was set as target and rest of the attributes were selected for the testing of trained model. The testing set have the five input features and one target label of # number of samples. The details of testing set are given in below table.

|  |  |  |
| --- | --- | --- |
| **Teachers Dataset** | Number of Samples in testing set | 2159 |
| Total attributes in testing set | 6 |
| Target Variable in testing set | Organization |

**Model Prediction -** Next the testing set was used for the prediction of service provider by each trained model. All the evaluation measures including the accuracy, precision, recall, and F1 score were calculated for testing sets. All the evaluation measures were calculated by using the built functions in Sklearn library of python.

**Result Analysis –** The selected evaluation measures were calculated on testing set to judge the best trained model for the prediction of best service provider. Among the built-in deep learning model, # model perform well in term of accuracy as it showed the highest accuracy. But the proposed model crossover all the model and outer perform by all other trained models. The customized model showed the 79% accuracy for prediction of best service provider. The score of evaluation measures of all model is also shown in below table. The comparative overview of all models accuracies is also presented in below figure.

Table : Evaluation Measures report of testing set.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy | Precision | Recall | F1 |
| TabNet | 0.7216 | 0.6791 | 0.7217 | 0.6801 |
| CNN | 0.7318 | 0.7205 | 0.7320 | 0.7014 |
| Tabular | 0.7253 | 0.8513 | 0.7253 | 0.7624 |
| DeepTables | 0.4506 | 0.2639 | 0.4505 | 0.3203 |
| 3 Layer MLP | 0.4279 | 0.2484 | 0.4280 | 0.3068 |
| Customized | 0.7890 | 0.7686 | 0.7890 | 0.7508 |

