**Big Data - Project Deliverable 3**

**Group 21**

**7. Analytics and Machine Learning:**

We have used AWS SageMaker Canvas for our data analytics part. The below are the steps performed in AWS SageMaker Canvas to perform some analytics on US Accidents Dataset.

1. **Loading the dataset:** We must import our dataset into SageMaker Canvas. Below is the screenshot showing the uploaded dataset into SageMaker.

Graphical user interface, table

Description automatically generated

The below screenshot shows the columns of the dataset and their properities like datatype, missing values, misplaced values, unique values, mean/mode value, correlation to the target. These values are automatically calculated after you upload the dataset.

Graphical user interface, application

Description automatically generated

The below image shows the grid view of the dataset attributes.

A screenshot of a computer

Description automatically generated

1. Building and training a ML model: Here you need to select a column to predict which is called a target column. For US accident dataset we have taken severity as the target column, and we try to predict the level of severity based on the attributes.

Then we must choose the appropriate model. SageMaker provides two different types of models by default, and they are time series forecasting and numeric model type. we have chosen the Numeric model.

1. Data transformation and preparation: Due to numerous problems, such as incomplete data or outliers, you might want to cleanup your data and do feature engineering to increase the precision of your model. You may clean, transform, and organize your information for model creation using the ML data transforms offered by Amazon SageMaker Canvas. Without writing any code, you may apply these changes to your datasets. The transformations you employ are added by SageMaker Canvas to the Models recipe, which serves as a log of the data preprocessing work carried out on your data prior to generating the model. Any data transformations you employ only change the input data used to develop models; they do not change the original data source.

A screenshot of a computer

Description automatically generated

The Model recipe gives the information about the operations that we performed on the dataset. The above screenshot shows that information. We have performed drop of columns on the dataset and made the date/time attributes look similar like changed the data type.

1. Visualizations: SageMaker also provides some visualization parts where we can choose the columns which we want to find the relation with and need to select the type of chart. The SageMaker will automatically generate the graph.

The below screenshot shows the box plot between Severity and Visibility attributes.

A screenshot of a computer

Description automatically generated

The below screenshot shows the bar graph for the severity and time zone attributes. The graph gives information about the severity level in different time zones.

Graphical user interface, application, Word

Description automatically generated

1. Model Building: The below screen shot shows the model building process. SageMaker provides us to the model building with just a click.

Graphical user interface, application, table, Excel

Description automatically generated

The below screenshot shows the estimated Root Mean Square Error of the model that we build. Our model got an RMSE of 0.361 which means the out of all the data there is a chance of 36% that our model goes wrong during prediction.

Also, the model often predicts a value that is within +/- 0.361 of the actual value for severity. Our model is not so bad, when prediction he severity. This error change is because of the categorical values in our dataset.

A screenshot of a computer

Description automatically generated

The below screenshot shows the attributes which has highest impact on the prediction. So, as per our dataset and numeric model, the state, start\_time, Start\_lat, End\_log, Humidity, start\_log and End\_Time attribues has highest impact on the severity prediction.

Graphical user interface, application, table

Description automatically generated

**8. Evaluation and Optimization:**

Though the 0.361 error rate is acceptable, we tried for better accuracy for predicting the severity level of the accident. So, we developed a neural network based on the dataset attributes. Neural network trains the model by keep on understanding the data at every iteration. This gives the better results in case of real time scenarios like US accidents

We have used Amazon SageMaker Jupyter Notebook for our analysis.

The below screenshot shows the importance of each attribute. For this we have used ExtraTreesRegressor model. Based on this analysis we can decide the columns that we want our Neural Network model should used for predicting the severity of accident.

From the below graph, we can understand that End\_Time, Start\_Time, End\_Lng, Start\_Lat, End\_Lat, Start\_Lng and distance attributes has a high impact on causing the accident.

Chart, bar chart

Description automatically generated

Then, we have created a Neural Network using python keras module. We have created several layers including relu and softmax layers. We have used adam optimizer and sparse\_categorical\_crossentropy as the loss function for our model. Adam and sparse categorical loss functions work well with categorical values. Also, we have used accuracy as the performance metric in our model.

The below screenshot shows the results for the model for 10 epochs.

Table

Description automatically generated

From the above results, we can say that our model works well as we got the accuracy of approximately above 95 at all the epochs. Which means that, the prediction that our model performed on severity of the accident is 95% correct. This accuracy is a must in case of real time scenarios like accident prediction.

The below screenshot shows the loss matrix. Which gives the information about actual loss and validation loss.

Table

Description automatically generated

The below is the graph for the actual loss and validation loss.

Chart, line chart

Description automatically generated

**9. Results:**

We have done two models for US accident dataset. One using the AWS SageMaker Canvas inbuilt model i,e numeric model. As the name suggests the numeric model performs well for numeric data or for the data which is of similar datatype. As the US accident dataset contains attributes with numeric, binary, categorical and text data types of attributes, the numeric model of SageMaker didn’t perform well. We got a Root Mean Square Error value of 0.361 which is acceptable in some case, but when we considered in real time cases like accidents, 36% error rate can lead to the un appropriate error. This may not be acceptable in real time accident prediction.

So, we performed a neural network which will try to analyze data at each iteration. Neural Network has the capability to adapt to new situations like new type of data addition to the dataset. Neural Network performs well with the data containing different types of attributes.

Our second model is a Neural Network model with adam optimizer, relu and softmax layers and soarse categorical cross entropy loss function. Also, we used accuracy as the evaluation metric for our prediction. The results of our Neural Network model show that our model works better with US accident dataset. We got an accuracy of 97% for predicting the severity of the accident.

**10. Future Work:**

**a. What was unique about the data?  Did you have to deal with imbalance? What data cleaning did you do? Outlier treatment?  Imputation?**

The most unique feature about US accident data is that the data contains categorical data. The attributes of the dataset are of different datatypes including text, string, numeric and so on.

We have noticed some imbalance in the dataset in terms of the datatypes for different attributes. For example, the attributes like Start\_Date and End\_Date has the date as their data type. But the values for these attributes are different something like MM/DD/YYYY or DD/MM/YYYY. This mismatch in data values will cause the imbalance in the data.

To solve this, we have did some data cleaning process using AWS SageMaker Canvas such as extracting the data values from the columns. For instance, we have performed the extract the End\_Date column and Start\_Date columns which makes the both the attributes has the same type of values.

**b. Did you create any new additional features / variables?**

No, we haven’t created any additional features / variables. As US accident dataset provides with all the required features that can cause an accident and that were required for our accident analysis. Also, there are no such attributes that were required to be combined with before the data processing. All the attributes were independent of each other.

**c. What was the process you used for evaluation?  What was the best result?**

We evaluated the data with the help of AWS SageMaker Canvas inbuilt evaluation methods like attributes outlier detection and so on. Also, we evaluated the data based on the plots that we created with different attributes. We created the different box, bar, and line plots with various attributes for understanding the relationship between the attributes. We found that some attributes have the inconsistent values like the date format for Start\_Date and End\_Date are different which leads to the data collision. So, we have extracted the data types from these two attributes for making them have the same data.

**d. Is there Bias in your work? What were the problems you faced? How did you solve them?**

We faced some problems with the SageMaker Canvas as we were new to the AWS concepts. But then we were able to solve them with the help of AWS discussions forum for solutions for the problems website. Also, we have some issues with the types of attributes during creating the model. Then we understood the reason for causing the error and preformed AWS SageMaker inbuilt extraction method for referring the datatypes of the attributes.

**e. What future work would you like to do?**

The future work for this project would include the following.

1. Finding the best algorithm which can predict the accident happening in real time scenarios with highest accuracy.
2. Developing a method for identifying the reason for causing the accident and doing the analysis based on this.