Business Problem

We would like you to come up with a solution approach for the following Challenge question. This will help us in shortlisting candidates for further process.

Please send your response by noon Feb 15th 2019 MT.

Challenge Questions:
In a maximum of one page, please answer the following two questions:
• What analytical techniques would you apply to mine insights from this data set?
• Using the maintenance/repair data attributes outlined above (and only those attributes), describe what actions XY Transport could take to predict failures and reduce unscheduled repairs.
Note: As you do not have actual data to work with, please state any assumptions incorporated into your challenge question response. Remember, there are no "right" answers to these questions, we simply want to see how you approach a business problem.

Below is data to transform the way XY Transport industry operates, we strongly encourage you to submit a 1-page response to the challenge questions above:
Assume you have a large data set (2-3 years' worth) of truck maintenance/repair data as follows:

Metadata:
- *Unique truck identifier*
- *truck mileage*
- *Maintenance/Repair date*
- *Maintenance/Repair indicator (i.e. Maintenance or Repair)*
- *Scheduled/Unscheduled indicator*
- *Maintenance/Repair category (Wheels, bearing, engine, internal system failure)*
- *Maintenance/Repair description*

Sample Data:
- *XY4326*
- *50000km*
- *Jan 3, 2019*
- *Maintenance*
- *Scheduled*
- *Internal System Failure*
- *PTC failure*

--
- *XY7399*
- *65000km*
- *Jan 1, 2019*
- *Repair*
- *Unscheduled*
- *Wheels*
- *Wheels replacement*

--
- *XY7399*
- *65000km*
- *Dec 24, 2018*
- *Repair*
- *Unscheduled*
- *Engine*
- *Air brake repair*

(Solution next page)

Overview of business problem

The goal of the problem is to reduce the number of unscheduled repairs. We have a very large dataset spanning 3 years. The assumption here is that the dataset is clean and ready for analysis. It is in the form of a Pandas dataframe. The “unique” truck identifier variable will be excluded from analysis. We would be using Python and Spark.

Exploratory Analysis

As we have the date field, we can perform a seasonal analysis. Let’s assume that the peak winter months are from December through March. We would slice rows where the Maintenance/Repair column reads ‘Repair’ and the ‘Scheduled/Unscheduled’ column reads ‘Unscheduled’. We could use resampling over a 1-2 week window by percentages of repair categories. This would enlighten us on the categories of repairs (that are unscheduled) that peak during the winter months. We could also attempt to find correlation between different repairs in a season. These steps could be repeated for the other seasons as well. We could also perform hypothesis testing using t-sample tests on the complete dataset to see if vehicles with higher mileage are more prone to undergo unscheduled repairs.

Potential Models for predicting failures ahead of time

We could use an LSTM (Long Short Term Memory) model since we have the dates. LSTM networks are state of the art architectures (by industry standards) for predicting upward and downward trends (as in the stock market). For this problem, we want to predict the upward and downward trends for unscheduled repairs for each month of the year 2018 given 2016 and 17 data.

We would have to perform some data preprocessing first. For each day, we would have to compute the total number of unscheduled repairs. Given our master data file in the Pandas dataframe format, we would create an additional column ‘Unscheduled Repairs’ that would indicate a ‘1’ if the scheduled/unscheduled indicator is ‘unscheduled’ and Maintenance/Repair indicator is ‘Repair’. The dataframe would then be sliced to include only the ‘Date’ and ‘Unscheduled Repair column’. We would then do a group by using the Maintenance/repair date and aggregate using .sum(). This would compute the total number of unscheduled repairs per day. Let’s call this new column as ‘URD’ (short for Unscheduled Repairs by day). For our analysis, we would be predicting URD, which is a continuous variable. In the next step of data preprocessing, we would have to split our data into test and train data sets. If we have 3 years’ worth of data, then, we can we can use the first 2 years of data as our training set and our last 1 year of data as our test set. Next, on the training set we can perform feature scaling using normalization on the URD column. This can be achieved using the MinMaxScaler which is available in the sklearn.preprocessing library. We could instantiate this object using the parameter (feature_range = (0,1)). We then use the fit_transform method on the ‘URD’ column.

Once the data preprocessing is complete, we would then go about building a data structure with the optimal number of time_steps so that we don’t overfit our model. We could also use dropout and regularization for robustness of our LSTM model. The model could be compiled using the “mean squared error” loss function and we would train the model using the adam optimizer.

Model results, evaluation and future steps

Our ultimate goal is to predict upward/downward trends on a monthly basis. Given the January months of 2016 and 2017, we train the model to predict the number of unscheduled repairs for January 2018. If there has been a surge in unscheduled repairs during early Jan through mid-January of both 2016 and 17, then we would expect our model to predict an upward trend in the number of unscheduled repairs from early to mid-Jan for 2018. This insight will better prepare the company to handle unscheduled repairs and using this timeline, the company can outline steps to counter unscheduled repairs. Now that we have our optimal model that performed well on the 2018 dataset, we could use the same model for 2019. In 2020, we could train and test the model using 4 years’ worth of data in which case we can expect the model to predict trends more efficiently.