**Machine Learning Engineer Nanodegree**

**Capstone Proposal**

Srinivasa Amirapu  
July 24th, 2018

**Proposal**

Telstra Network Disruptions

(https://www.kaggle.com/c/telstra-recruiting-network)

The goal of the problem is to predict Telstra network's fault severity at a time at a particular location based on the log data available. Using a dataset of features from their service logs, you're tasked with predicting if a disruption is a momentary glitch or a total interruption of connectivity.

**Domain Background**

Telstra is on a journey to enhance the customer experience - ensuring everyone in the company is putting customers first. In terms of its expansive network, this means continuously advancing how it predicts the scope and timing of service disruptions. Telstra wants to see how you would help it drive customer advocacy by developing a more advanced predictive model for service disruptions and to help it better serve its customers.

**Problem Statement**

This is a multi-class classification problem to predict 3 categories of faults: 0,1,2 (0 meaning no fault, 1 meaning only a few, and 2 meaning many).

The goal of the problem is to predict Telstra network's fault severity at a time at a particular location based on the log data available. Using a dataset of features from their service logs, you're tasked with predicting if a disruption is a momentary glitch or a total interruption of connectivity.

**Datasets and Inputs**

Datasets can be downloaded from Kaggle: https://www.kaggle.com/c/telstra-recruiting-network/data

•train.csv - the training set for fault severity

•test.csv - the test set for fault severity

•sample\_submission.csv - a sample submission file in the correct format

•event\_type.csv - event type related to the main dataset

•log\_feature.csv - features extracted from log files

•resource\_type.csv - type of resource related to the main dataset

•severity\_type.csv - severity type of a warning message coming from the log

Each row in the main dataset (train.csv, test.csv) represents a location and a time point. They are identified by the "id" column, which is the key "id" used in other data files. Fault severity has 3 categories: 0,1,2 (0 meaning no fault, 1 meaning only a few, and 2 meaning many).Different types of features are extracted from log files and other sources: event\_type.csv, log\_feature.csv, resource\_type.csv, severity\_type.csv.Note: “severity\_type” is a feature extracted from the log files (in severity\_type.csv). Often this is a severity type of a warning message coming from the log. "severity\_type" is categorical. It does not have an ordering. “fault\_severity” is a measurement of actual reported faults from users of the network and is the target variable (in train.csv).

**Solution Statement**

The goal of the problem is to predict Telstra network's fault severity at a time at a particular location based on the log data available.

For training models I will compare SVC,LogisticRegression,GradientBoostClassifier since this is a multi-class classification problem. Finally I will select the best model for this problem and fine tune parameters to get best accuracy by using gridSearchCV.

**Benchmark Model**

**Algorithm Selection**

A wide class of models can be used for Network Disruption Prediction

▪ Generalized Linear Models

▪ Support Vector Machines

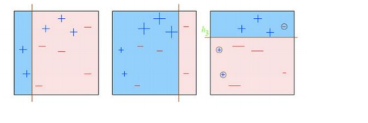
▪ K-Nearest Neighbors

▪ Naïve Bayes

▪ Decision Trees and Ensemble Methods

Among Ensemble Methods, we have chosen Gradient Boosting Classifier as benchmark model because of its better performance and accuracy.

Boosting is a sequential technique which works on the principle of ensemble. It combines a set of weak learners and delivers improved prediction accuracy.

**

The overall parameters can be divided into 3 categories:

 Tree-Specific Parameters: These affect each individual tree in the Model

Min\_samples\_split, min\_samples\_leaf,

min\_weight\_fraction\_leaf, max\_depth, max\_leaf\_nodes,

max\_features

 Boosting Parameters: These affect the boosting operation in the model

Learning\_rate, n\_estimators, subsample

 Miscellaneous Parameters: Other parameters for overall functioning Loss, init, random\_state, verbose, warm\_start, presort

**Evaluation Metrics**

*Model accuracy can be evaluated using the multi-class logarithmic loss. Each data row has been labeled with one true class. For each row, you must submit a set of predicted probabilities (one for every fault severity). The formula is then,*

*logloss=−1N ∑ i=1 N ∑ j=1 M y ij log(p ij ),*

*logloss=−1N∑i=1N∑j=1Myijlog⁡(pij),*

*where N is the number of rows in the test set, M is the number of fault severity classes, log*

*log is the natural logarithm, y ij is 1 if observation i*

*belongs to class jj and 0 otherwise, and p ij*

*pij is the predicted probability that observation i belongs to class j*

*The submitted probabilities for a given row are not required to sum to one because they are rescaled prior to being scored (each row is divided by the row sum). In order to avoid the extremes of the log function, predicted probabilities are replaced with max(min(p,1−10 −15 ),10 −15 )*

**Project Design**

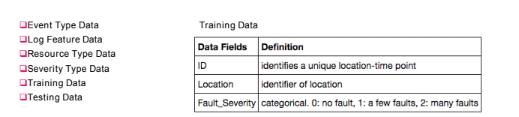
***Programming language****: Python 3.0*

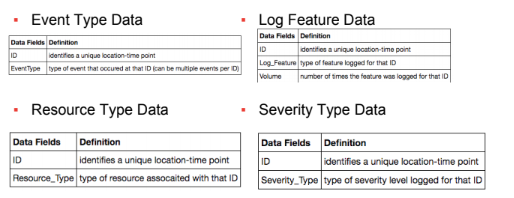
***Libraries*** *:Scikit-Learn*

***WorkFlow****:*

**1)Data Understanding**

*The data set is in a relational format, split among multiple files. The following provides a description of data in each file.*

**

**

**2) Data Preparation**

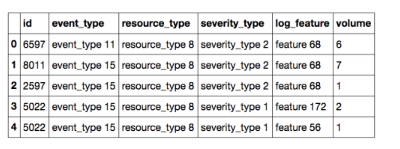
Step 1: Import Modules

Step 2: Import Datasets

Each row in the main dataset (train.csv, test.csv) represents a location and a time point. They are identified by the "id" column, which is the key "id" used in other data files. Fault severity has 3 categories: 0,1,2 (0 meaning no fault, 1 meaning only a few, and 2 meaning many). “fault\_severity” is a measurement of actual reported faults from users of the network and is the target variable (in train.csv).

Step 3: Data preprocessing

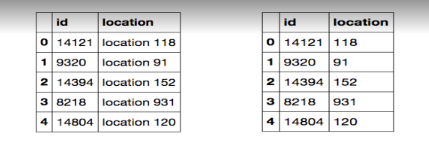
Step 4: Data merging to create a single record



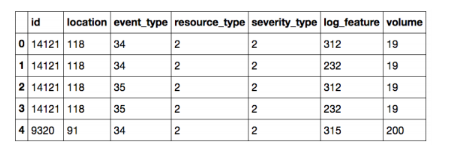
Step 5: Remove text from variables

Features are extracted from log files and other sources:event\_type.csv, log\_feature.csv, resource\_type.csv,severity\_type.csv. All above features are categorical except for "volume".

Step 6: Drop ”fault\_severity” from train dataset as it is the target variable

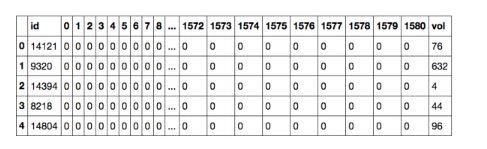


Step 7: Merge the train dataframe without the “fault\_severity” column and the combined dataframe of “event\_type ... etc”



Step 8: Convert Categoricals using Get\_Dummies

Step 9: groupby “id”



**3) Split data** using train\_test\_split method from sklearn.model\_selection

**4) Model Selection**: Choose the best model with best accuracy (fbeta\_score) and performance. Among Ensemble Methods, we have chosen Gradient Boosting Classifier as benchmark model because of it generally has better performance and accuracy.

**5) Model Training:** Train the gradient boost classifier model using optimal parameters (best fit model) obtained from GridSearchCV technique.

**6)** **Predict:** Calculate predicted values along with probabilities for test data set.

**7) Measure accuracy:** CalculatePrecision Recall metrics for multi-class classification problem.