CMPE-256 LARGE SCALE ANALYTICS

SPRING-2018

Project Report

Presented to:

Professor Sanket Tavarageri

Presented By:

Monank Savaliya

Umang Kakaiya

Vishweshkumar Patel

Date of Submission: 04/25/2018

1. Project Description:

Description:

Predicting “Call Type” for Fire Department Calls in city of San Francisco based on historical records from Year 2000 to Year 2018.

Goal and Use-cases:

The project focuses to use historical records from Year 2000 to Year 2018 of Fire Department Calls of city of San Francisco to predict(classify) “Call Type” for a specific location for certain day and time of week.

The main goal of the project is to predict(classify) “Call Type” from four different types - Fire, Alarm, Potential Life Threatening and Non-Life Threatening for a call made to Fire Department of San Francisco city.

The use-case will provide more insights to respective fire departments to take important decisions to improve their services and save more lives by predicting call type for future calls. It will be useful to decide which type of emergency situation it is and which kind of help would be more helpful.

2. Dataset Description

Dataset SourceType:
Public Dataset from “Fire Department Calls For Service”

Source:
https://data.sfgov.org/Public-Safety/Fire-Department-Calls-forService/nuek-vuh3/data

Data Dictionary:
https://data.sfgov.org/api/views/nuek-vuh3/files/ddb7f3a90160-4f07-bb1e-2af744909294?download=true&filename=FIR0002_DataDictionary_fire-calls-for-service.xlsx

Attribute Details:

Field Name	Data Type	Definition
Call Number	Text	A unique 9-digit number assigned by the 911 Dispatch Center (DEM) to this call. These number are used for both Police and Fire calls.
Unit ID	Text	Unit Identifier. For example E01 for Engine 1 or T01 for Truck 1.
Incident Number	Text	A unique 8-digit number assigned by DEM to this Fire incident.
Call Date	Date & Time	Date the call is received at the 911 Dispatch Center. Used for reporting purposes.
Call Type	Text	Type of call the incident falls into. See the list below.
Watch Date	Date & Time	Watch date when the call is received. Watch date starts at 0800 each morning and ends at 0800 the next day.
Received DtTm	Date & Time	Date and time of call is received at the 911 Dispatch Center.

Entry DtTm	Date & Time	Date and time the 911 operator submits the entry of the initial call information into the CAD system
Dispatch DtTm	Date & Time	Date and time the 911 operator dispatches this unit to the call.
Response DtTm	Date & Time	Date and time this unit acknowledges the dispatch and records that the unit is en route to the location of the call.
On Scene DtTm	Date & Time	Date and time the unit records arriving to the location of the incident
AVL Validated On Scene DtTm	Date & Time	Updated date and time the unit arrived on scene based on existing unit AVL coordinates.
Transport DtTm	Date & Time	If this unit is an ambulance, date and time the unit begins the transport unit arrives to hospital
Hospital DtTm	Date & Time	If this unit is an ambulance, date and time the unit arrives to the hospital.
Call Final Disposition	Text	Disposition of the call (Code). For example TH2: Transport to Hospital - Code 2, FIR: Resolved by Fire Department
Available DtTm	Date & Time	Date and time this unit is not longer assigned to this call and it is available for another dispatch.
Address	Text	Address of midblock point associated with incident (obfuscated address to protect caller privacy)
City	Text	City of incident
Zip Code of Incident	Text	Zip Code of incident
Battalion	Text	Emergency Response District (There are 9 Fire Emergency Response Districts)

Station Area	Text	Fire Station First Response Area associated with the address of the incident
Box	Text	Fire box associated with the address of the incident. A box is the smallest area used to divide the City. Each box is associated with a unique unit dispatch order. The City is divided into more than 2,400 boxes.
Original Priority	Text	Initial call priority (Code 2: Non-Emergency or Code 3:Emergency).
Priority	Text	Call priority (Code 2: Non-Emergency or Code 3:Emergency).
Final Priority	Text	Final call priority (Code 2: Non-Emergency or Code 3:Emergency).
ALS Unit	Boolean (True/False)	Does this unit includes ALS (Advanced Life Support) resources? Is there a paramedic in this unit?
Call Type Group	Text	Call types are divided into four main groups: Fire, Alarm, Potential Life Threatening and Non Life Threatening.
Number of Alarms	Numeric	Number of alarms associated with the incident. This is a number between 1 and 5.
Unit Type	Text	Unit type
Unit sequence in call dispatch	Numeric	A number that indicates the order this unit was assigned to this call
Location	Coordinates	Latitude and longitude of address obfuscated either to the mid block, intersection or call box
Fire Prevention District	Text	Bureau of Fire Prevention District associated with this address
Supervisor District	Text	Supervisor District associated with this address

Neighborhood District	Text	Neighborhood District associated with this address, boundaries available here: https://data.sfgov.org/d/p5b7-5n3h
RowID	Text	Unique identifier used for managing data updates. It is the concatenation of Call Number and Unit ID separated by a dash

Dataset Size:
Rows: 4.61 million
Columns: 34
Size in GB: 1.5 GB

Sample Data:

Call Number	1030101	1030104	1030106	1030107
Unit ID	E18	M14	M36	E01
Incident Number	306091	30612	30614	30615
Call Type	Medical Incident	Medical Incident	Medical Incident	Alarms
Call Date	04-12-00	04-12-00	04-12-00	04-12-00
Watch Date	04-12-00	04-12-00	04-12-00	04-12-00
Received DtTm	########	########	########	########
Entry DtTm	########	########	########	########
Dispatch DtTm	########	########	########	########
Response DtTm		########		########

On Scene DtTm		########	########	########
Transport DtTm		########	########	
Hospital DtTm		########	########	
Call Final Disposition	Other	Other	Other	Other
Available DtTm		########	########	########
Address	2000 Block of 37TH AVE	1700 Block of 43RD AVE	0 Block of FELL ST	100 Block of JONES ST
City	SF	SF	SF	SF
Zipcode of Incident	94116	94122	94102	94102
Battalion	B08	B08	B02	B03
Station Area	18	23	36	1
Box	757	7651	3111	1456
Original Priority	3	3	3	3
Priority	3	3	3	3
Final Priority	3	3	3	3
ALS Unit	FALSE	TRUE	FALSE	FALSE
Call Type Group				
Number of Alarms	1	1	1	1
Unit Type	ENGINE	MEDIC	MEDIC	ENGINE

Unit sequence in call dispatch	1	2	1	3
Fire Prevention District	8	8	2	3
Supervisor District	4	4	6	6
Neighborhoods - Analysis Boundaries	Sunset/Parkside	Sunset/Parkside	Tenderloin	Tenderloin
Location	(37.7487247711275, -122.495504020186)	(37.754032678059 5, -122.502185504543)	(37.776440510083 8, -122.418481123408)	(37.782547400042 1, -122.412247935495)
RowID	001030101-E18	001030104-M14	001030106-M36	001030107-E01

3. Data Cleaning

Data PreProcessing & Data Exploration Phase:

- According to our use case, found out irrelevant features and removed them, such as - 'call_number', 'unit_id', 'rowid' etc.

- Attribute such as
  - 'city' contains values with different representation for same values such as 'San Francisco', 'SAN FRANCISCO', 'SF' for city of San Francisco. So we have retained only single representation for these values.
  - 'supervisor_district', 'box' and 'station_area' contains string representation as well as numeric representation for same values. So we have only retained numeric representation for 'supervisor_district', 'box' and 'station_area'.

- Removed features which can only be known after providing response to a Call and do not contribute to predict 'call_type' - such as 'dispatch_dttm', 'entry_dttm', 'hospital_dttm', 'on_scene_dttm', 'response_dttm', 'transport_dttm', 'available_dttm', 'call_final_disposition', 'final_priority', 'priority', 'als_unit', 'unit_type', 'unit_sequence_in_call_dispatch'.

- Removed repetitive features (duplicate features) such as
  - 'call_date', 'watch_date' which can be combinely represented using 'received_dttm'.

- Considered following 14 attributes for further exploration:
  - 'Incident_number'
  - 'call_type_group'
  - 'received_dttm'
  - 'address'
  - 'zipcode_of_incident'
  - 'battalion'
  - 'station_area'
  - 'box'
  - 'original_priority'

    - 'number_of_alarms'
    - 'location'
    - 'fire_prevention_district'
    - 'supervisor_district'
    - 'neighborhoods_analysis_boundaries'

- Remove the duplicate instances involved in refined dataset after removing the irrelevant features.
- Found out the missing values, null values for remaining 14 attributes and removed those tuples. After removing those tuples still we have nearly 2 Millions of tuples remaining to train model.

4. Feature Engineering

In Feature Extraction and Engineering step, from available features we extract new features using dimensionality reduction techniques – PCA, SVD, etc OR feature engineering techniques - Date and time features, numeric to categorical mappings, grouping sparse classes, etc. that are more suitable for prediction task to achieve maximum accuracy.

- Removed all the duplicate rows for remaining 14 features.
- Removed all the rows with null values.
- Identified that 'Box' and 'Supervisor District' includes float, int and string data types making it hard for using the feature. Converted all instances of 'Box' and 'Supervisor District' into string using user defined function.
- Converted 'Received DtTm' feature into weekday, hour and month using user defined function. Defined three functions namely convert_date_to_weekday, convert_date_to_hour, convert_date_to_month.
- Utilized 'Station Area' feature initially, but due to sparse distribution and less variable importance of Station Area, we finally removed the feature for model fitting.
- Using "Address", "Box", "Month", "Weekday", "Hour", and "Neighborhoods_analysis_boundaries" created a CSR sparse matrix that can efficiently utilize the memory.
- Applied dimensionality reduction technique namely "TruncatedSVD" on CSR matrix.
- Plotted explained variance versus number of components graph and to figure out inflexion point and to choose right components.
- Divided the dataset into training and testing dataset.

5. Machine Learning Models

In Model(s) Training and Testing step, we selected appropriate models for classification tasks – Random Forest, XGBoost, Neural Network can be trained using transformed features obtained through feature engineering step. Dataset will be divided in Training and Testing datasets. We applied following machine learning models:

1. Random Forest
    - Random Forest is an ensemble learning model for classification, operating by making multitude of decision trees. It uses feature bagging concept to form series of decision trees by selecting random features of input dataset. For classification we take the majority votes given by the decision trees.
    - Used Sklearn package for Random Forest and trained the model with following parameter:
        - max_depth=5
        - Random_state = 42

2. MLP classifier
    - MLP (Multilayer perceptron) is a feed forward artificial neural network.
    - MLP consists of at least three or more layers with activation function added to each layer. Learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result.
    - We train the model with parameters as follows:
        1. Solver: lbfgs (family of quasi-Newton methods)
        2. Alpha: 1 e^{-5}
        3. HIdden Layer Tuple: (10, 10)

3. XGBoost (Extreme Gradient Boosting)
    - XGBoost is an implementation of gradient boosted decision trees designed for speed and performance.
    - It is a tree ensemble model. The tree ensemble model is a set of classification and regression trees (CART).

- Boosting is an ensemble method where new models are added to correct the errors made by existing model. Models are added sequentially until no further improvements can be made.
- We train the model with parameters as follows:
    1. Number of estimators: 50
    2. Max_depth: 4

4. SVM (Support Vector Machine)
    - Type of Discriminative classifier formally defined by a separating hyperplane. Applied SVC (Support Vector Classification) using Sklearn package with default parameters as C=1 and Degree= 3.

6. Results:

- To check accuracy of our trained models we divided entire data from year 2000 to year 2018 into training and testing. The proportion of training and testing dataset is defined below:

- Train data: (year 2000 to year 2015)
- Test data: (year 2016 to 2018) :

Model Name	Training Accuracy (F-1 Score)	Testing Accuracy (F-1 Score)
MLP Classifier	0.71	0.759
Random Forest Classifier	0.71	0.76
XGBoost Classifier	0.71	0.759
SVM Classifier	0.71	0.759

7. Technical Challenges Faced

- Due to the real life dataset collected from SF Fire Department website, the need for preprocessing was crucial. Data preprocessing required precise understanding of importance of features and what kind of input classifier accepts.

- Due to the large size of the dataset - 2.08 million rows, we couldn't apply one hot encoding directly to all the categorical variables. So, We had to utilize CSR matrix to handle the large sized sparse data.

- Even after converting the rows into CSR matrix form we couldn't directly provide it to model. So, we had to apply dimensionality reduction technique namely "TruncatedSVD". And choosing the right number of components was a big challenge in this technique, with consideration of preserving the characteristics of sparse data.

- Even though applying different models and choosing different combinations of features with the different number of component selection from "TruncatedSVD", we were not able to tune the model.