In [1]:

```
# !pip install pycountry_convert
# !pip install pandas-profiling
# !pip install pydot
# !pip install scikit-optimize
# !pip install lightgbm
```

```
Requirement already satisfied: pycountry_convert in /home/ray080186/.pyenv/versions/3.9.0/li
b/python3.9/site-packages (0.7.2)
Requirement already satisfied: pytest-cov>=2.5.1 in /home/ray080186/.pyenv/versions/3.9.0/li
b/python3.9/site-packages (from pycountry_convert) (4.0.0)
Requirement already satisfied: pprintpp>=0.3.0 in /home/ray080186/.pyenv/versions/3.9.0/lib/
python3.9/site-packages (from pycountry_convert) (0.4.0)
Requirement already satisfied: pytest-mock>=1.6.3 in /home/ray080186/.pyenv/versions/3.9.0/l
ib/python3.9/site-packages (from pycountry_convert) (3.10.0)
Requirement already satisfied: pycountry>=16.11.27.1 in /home/ray080186/.pyenv/versions/3.9.
0/lib/python3.9/site-packages (from pycountry_convert) (22.3.5)
Requirement already satisfied: wheel>=0.30.0 in /home/ray080186/.pyenv/versions/3.9.0/lib/py
thon3.9/site-packages (from pycountry_convert) (0.38.4)
Requirement already satisfied: pytest>=3.4.0 in /home/ray080186/.pyenv/versions/3.9.0/lib/py
thon3.9/site-packages (from pycountry_convert) (7.2.1)
Requirement already satisfied: repoze.lru>=0.7 in /home/ray080186/.pyenv/versions/3.9.0/lib/
python3.9/site-packages (from pycountry_convert) (0.7)
Requirement already satisfied: coverage[toml]>=5.2.1 in /home/ray080186/.pyenv/versions/3.9.
0/lib/python3.9/site-packages (from pytest-cov>=2.5.1->pycountry_convert) (7.2.1)
Requirement already satisfied: setuptools in /home/ray080186/.pyenv/versions/3.9.0/lib/pytho
```

In [4]:

```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
import seaborn as sns
from pandas_profiling import ProfileReport
import pycountry_convert as pc
from sklearn.preprocessing import StandardScaler, PolynomialFeatures
from sklearn.feature_selection import SelectKBest, mutual_info_regression, f_regression
from sklearn.pipeline import Pipeline, make_pipeline
from sklearn.model_selection import cross_validate, GridSearchCV, train_test_split
from sklearn.linear_model import LinearRegression, Ridge, Lasso
from sklearn.neural_network import MLPRegressor
from statsmodels.regression.linear_model import OLS
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import export_graphviz
import itertools
import pydot
from statsmodels.api import add_constant
from IPython.display import Image
from skopt import BayesSearchCV
import lightgbm as lgb
```

Question 1.1

Plot a heatmap of the Pearson correlation matrix of the dataset columns. Report which features have the highest absolute correlation with the target variable. In the context of either dataset, describe what the correlation patterns suggest.

> Ans: Pearson correlation values range from -1 to +1. The absolute value indicates how strongly the two variables relate to each other, while the sign reveals whether they are positively or negatively related. From the heatmap, we see that features 'x', 'y', 'z' and 'carat' have the highest absolute correlation values with the target variable 'price'. Hence, we can conclude that 'x', 'y', 'z', and 'carat' are the most informative information if we want to predict the diamond price.

In [5]:

```
diamonds_original = pd.read_csv('diamonds.csv')
diamonds_original = diamonds_original.drop(columns=['Unnamed: 0'])
```

In [6]:
```
cut_numeric = {'Fair': 1, 'Good': 2, 'Very Good': 3, 'Premium': 4, 'Ideal': 5}
color_numeric = {'J': 1, 'I': 2, 'H': 3, 'G': 4, 'F': 5, 'E': 6, 'D': 7}
clarity_numeric = {'I1': 1, 'SI2': 2, 'SI1': 3, 'VS2': 4, 'VS1': 5, 'VVS2': 6, 'VVS1': 7,
diamonds_original['cut_numeric'] = diamonds_original.cut.map(cut_numeric)
diamonds_original['color_numeric'] = diamonds_original.color.map(color_numeric)
diamonds_original['clarity_numeric'] = diamonds_original.clarity.map(clarity_numeric)
```

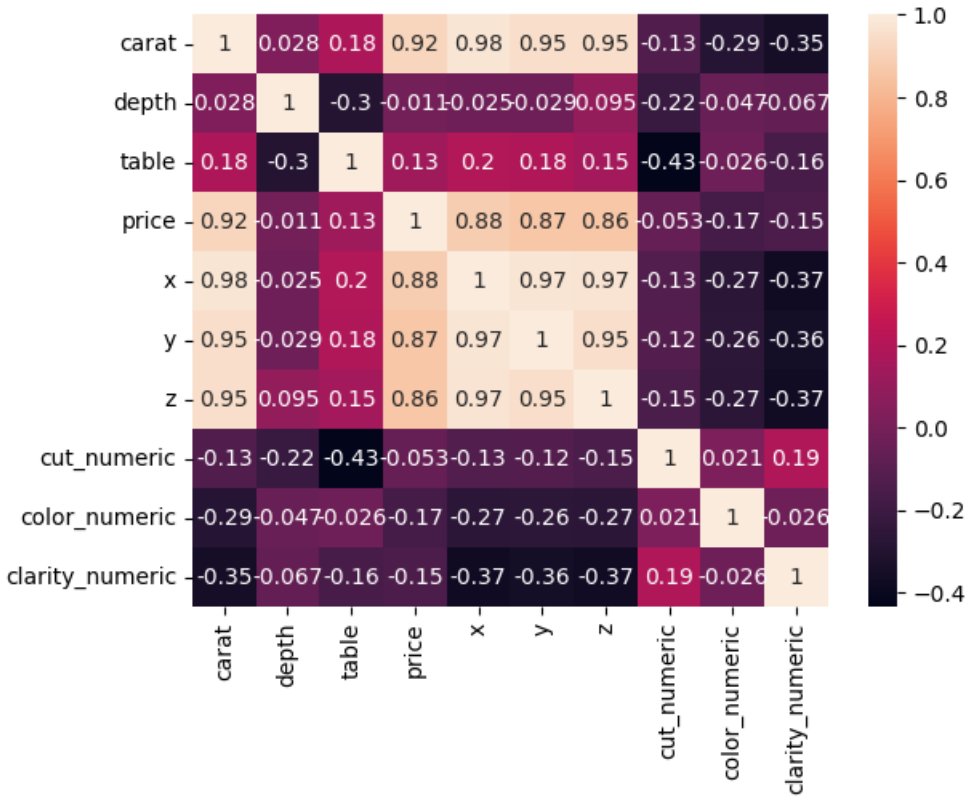
In [7]:
```
diamonds_original.head()
```

Out[7]:

	carat	cut	color	clarity	depth	table	price	x	y	z	cut_numeric	color_numeric	clarity_numeric
0	0.23	Ideal	E	SI2	61.5	55.0	330	3.95	3.98	2.43	5	6	2
1	0.21	Premium	E	SI1	59.8	61.0	327	3.89	3.84	2.31	4	6	3
2	0.23	Good	E	VS1	56.9	65.0	328	4.05	4.07	2.31	2	6	5
3	0.29	Premium	I	VS2	62.4	58.0	337	4.20	4.23	2.63	4	2	4
4	0.31	Good	J	SI2	63.3	58.0	338	4.34	4.35	2.75	2	1	2

In [8]:
```
diamonds_data = diamonds_original.copy()
diamonds_data = diamonds_data.drop(columns=['cut', 'color', 'clarity'])
corr = diamonds_data.corr()
sns.heatmap(corr, annot=True)
```

Out[8]: <AxesSubplot: >



Question 1.2

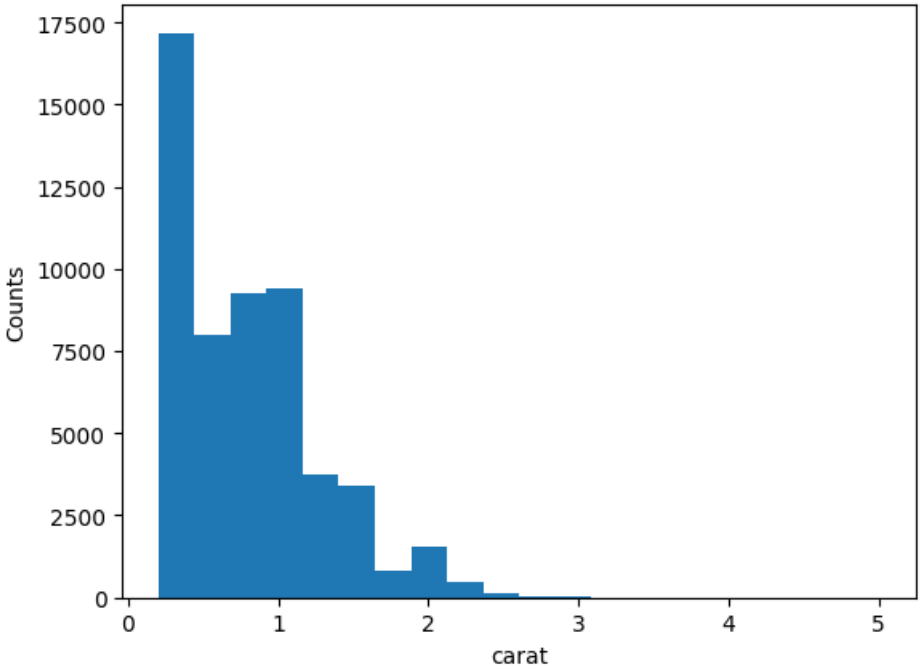
Plot the histogram of numerical features. What preprocessing can be done if the distribution of a feature has high skewness?

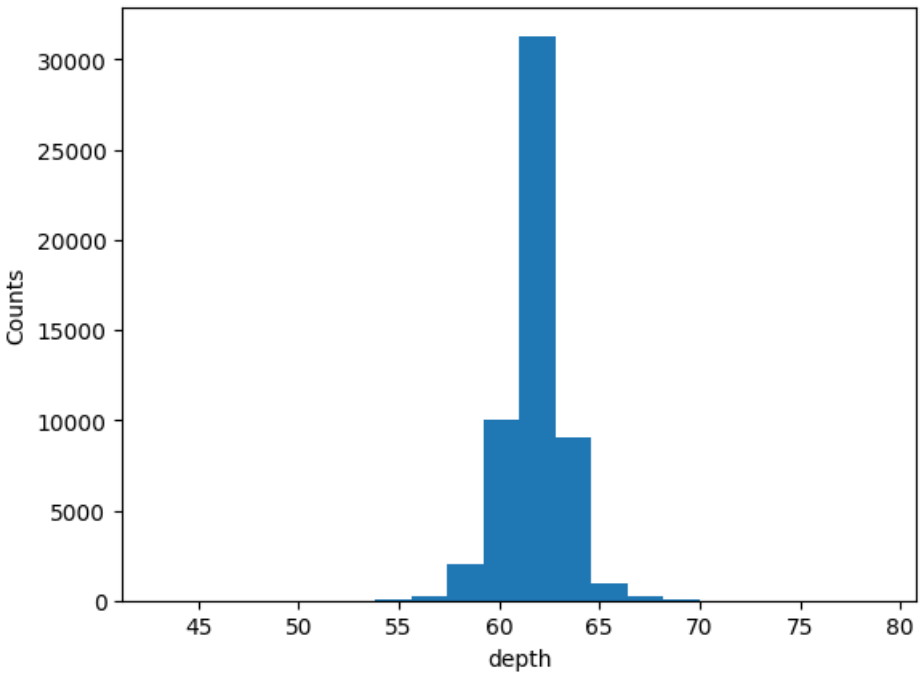
> Ans: From the histograms, we noticed that features 'carat' and 'price' have skewed distributions. Preprocessing with standardization, log transformation, and square root transformation etc. can help deal with skewness.

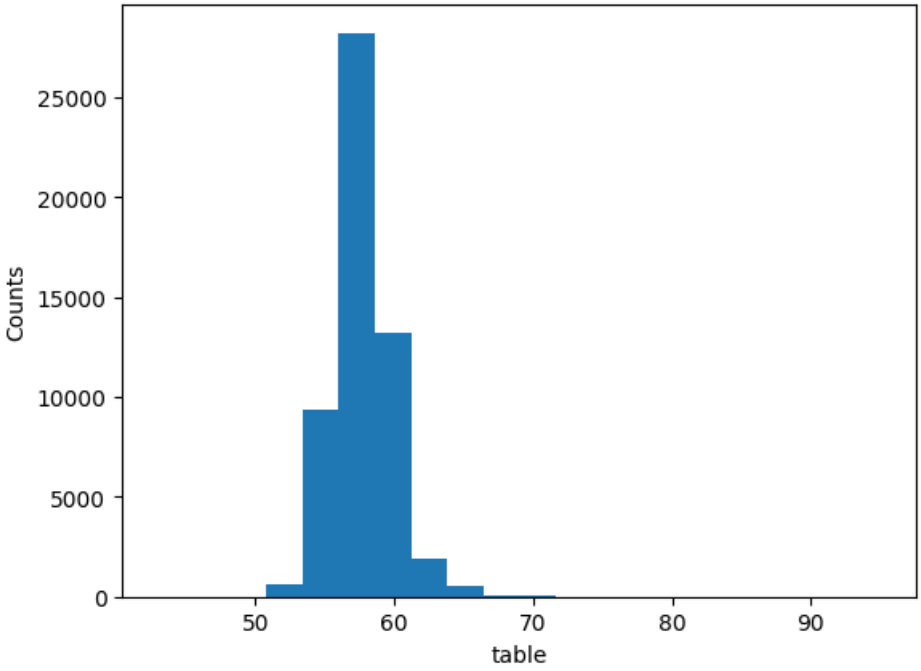
There are 7 numerical features: carat, depth, table, price, x, y, z. The distribution of each feature is shown as follows.

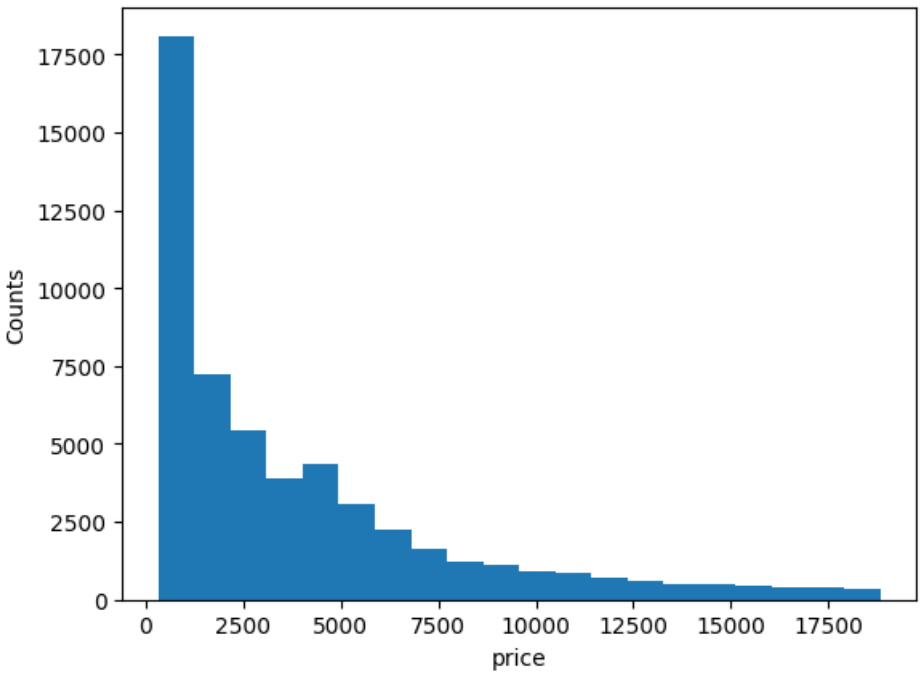
In [9]:

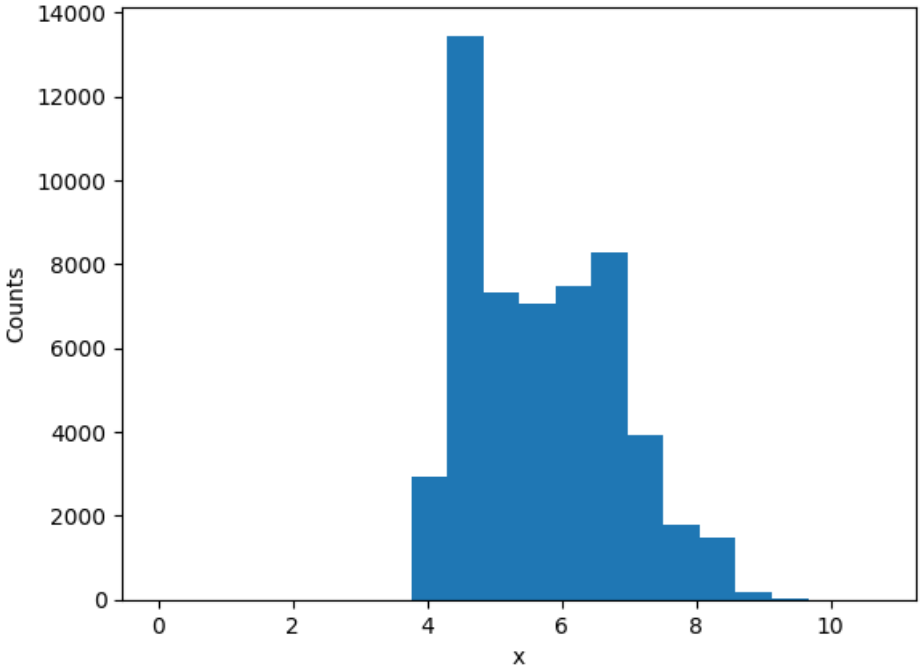
```
numerical_features = ['carat', 'depth', 'table', 'price', 'x', 'y', 'z']
for i in numerical_features:
    plt.hist(diamonds_original[i], bins=20)
    plt.xlabel(i)
    plt.ylabel('Counts')
    plt.show()
```

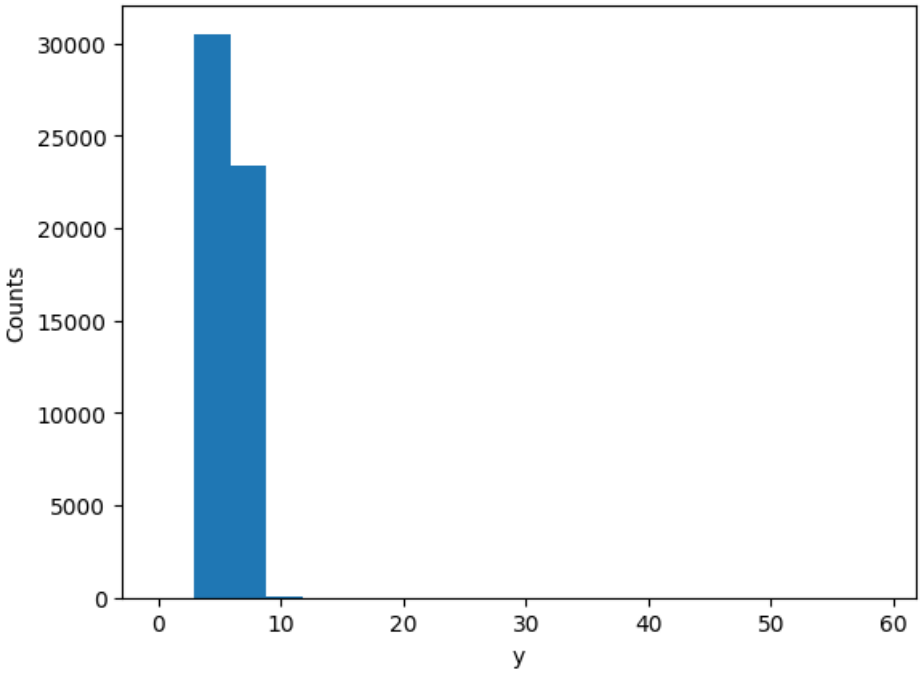


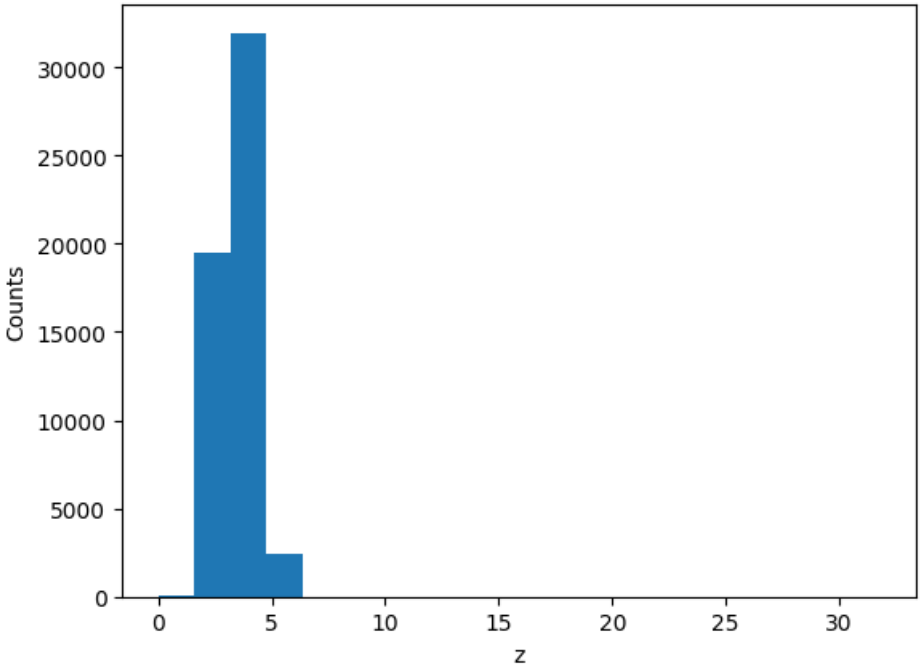



Counts
25000
20000
15000
10000
5000
0
50
60
70
80
90
table


Counts
17500
15000
12500
10000
7500
5000
2500
0
0
2500
5000
7500
10000
12500
15000
17500
price


14000
12000
10000
8000
6000
4000
2000
0
Counts
0
2
4
6
8
10
x


30000
25000
20000
15000
10000
5000
0
Counts
0
10
20
30
40
50
60
y



Question 1.3

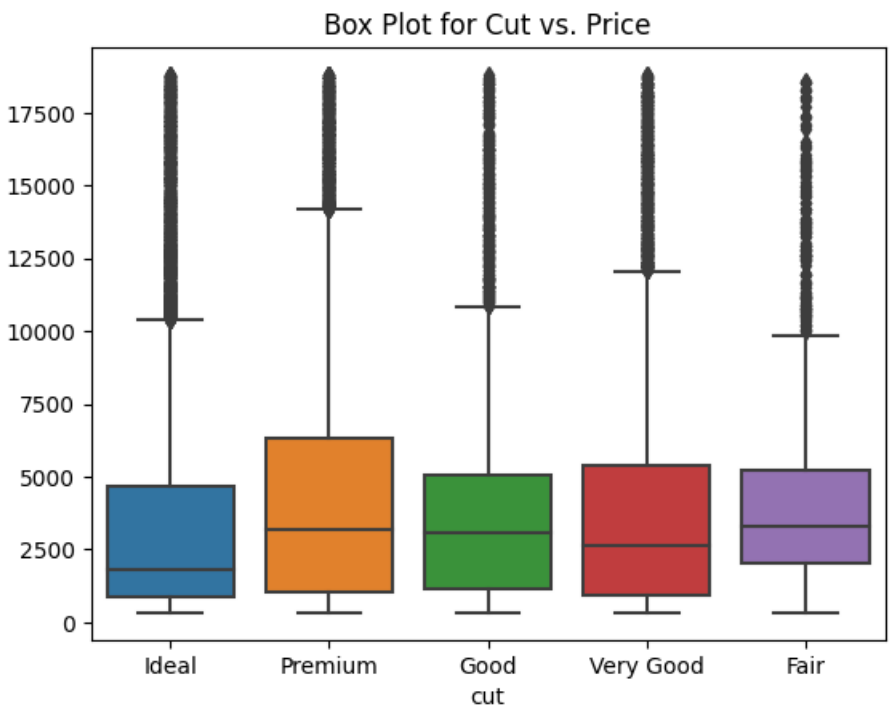
Construct and inspect the box plot of categorical features vs target variable. What do you find?

> Ans: There are totally three categorical features as the boxplots are shown below. For the `cut` feature, except that diamond with "Premium" cut tend to have slightly higher prices, we observed that the price distributions are similar given the the cut of the diamond. As for the `color` feature, we found that the price increases with the order of the character of the color code. Finally for the `clarity` feature, the distributions are mostly varied except two groups: 1. "IF" and "WS1", 2. "VS1" and "VS2".

There are 3 categorical features: cut, color and clarity. Each box plot is shown as follows.

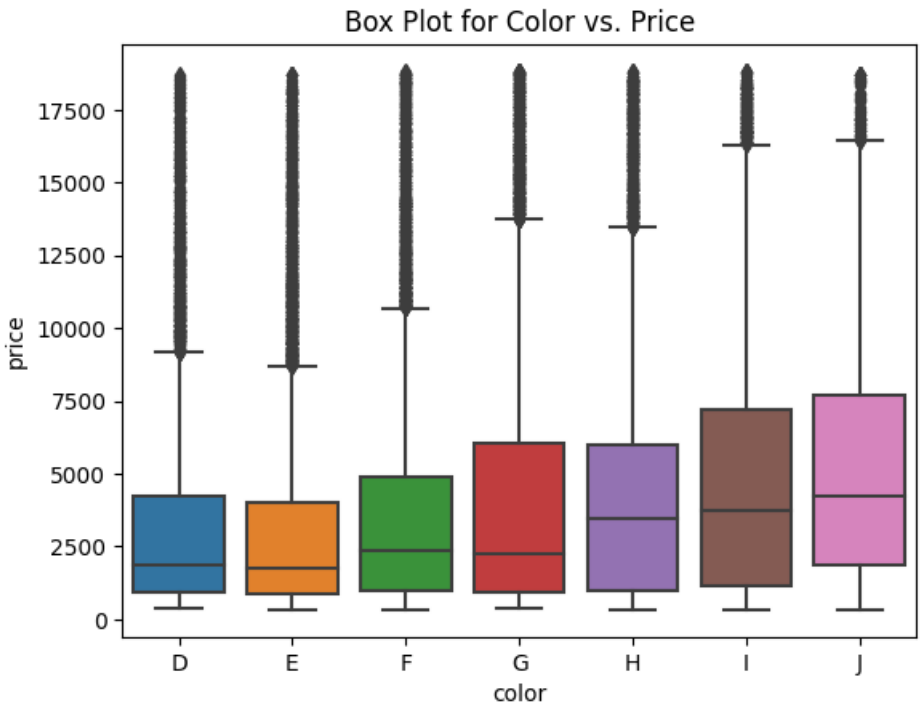
In [53]:
```
sns.boxplot(x=diamonds_original['cut'], y=diamonds_original['price'])
plt.title('Box Plot for Cut vs. Price')
```

Out[53]: `Text(0.5, 1.0, 'Box Plot for Cut vs. Price')`



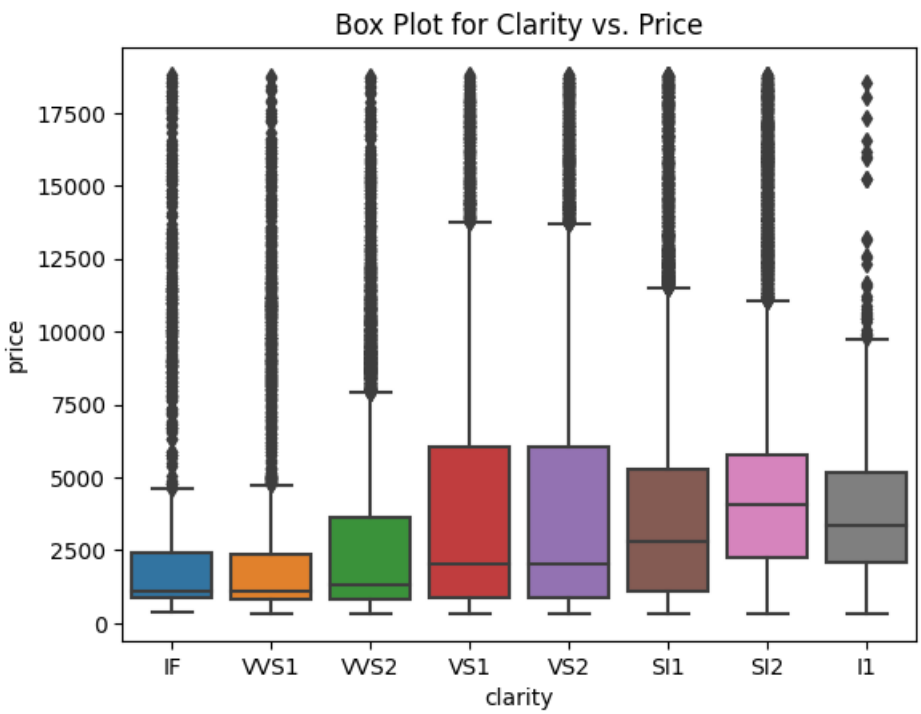
In [11]:
```
sns.boxplot(x=diamonds_original['color'], y=diamonds_original['price'], order=['D', 'E', '
plt.title('Box Plot for Color vs. Price')
```

Out[11]: Text(0.5, 1.0, 'Box Plot for Color vs. Price')



In [12]:
```
sns.boxplot(x=diamonds_original['clarity'], y=diamonds_original['price'], order=['IF', 'VV
plt.title('Box Plot for Clarity vs. Price')
```

Out[12]: Text(0.5, 1.0, 'Box Plot for Clarity vs. Price')

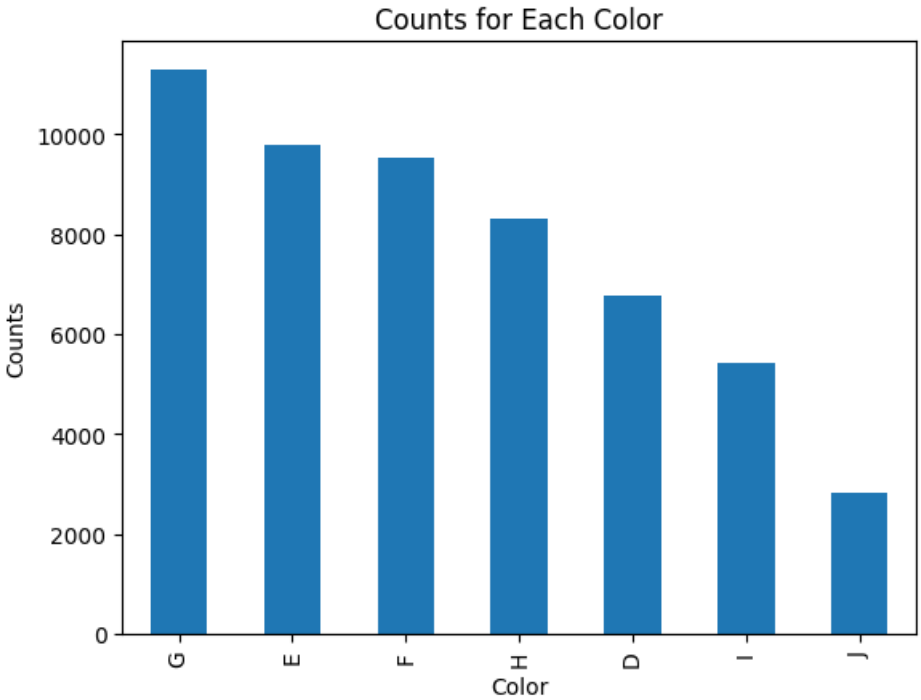


Question 1.4

For the Diamonds dataset, plot the counts by color, cut and clarity.

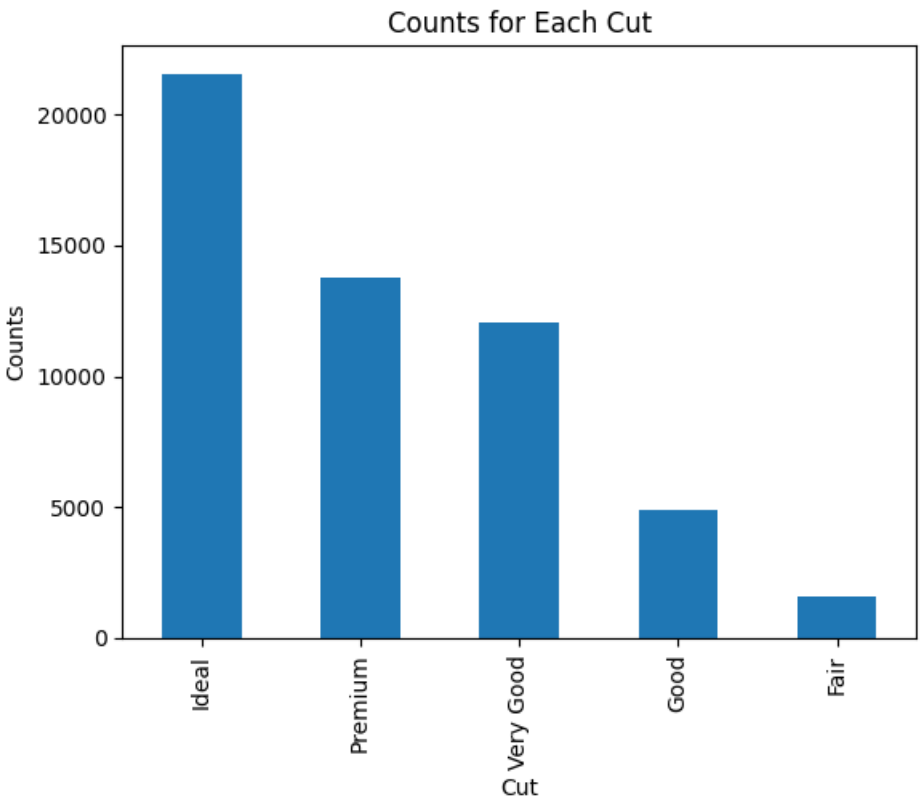
In [11]:
```
diamonds_original['color'].value_counts().plot(kind='bar')
plt.xlabel('Color')
plt.ylabel('Counts')
plt.title('Counts for Each Color')
```

Out[11]: Text(0.5, 1.0, 'Counts for Each Color')



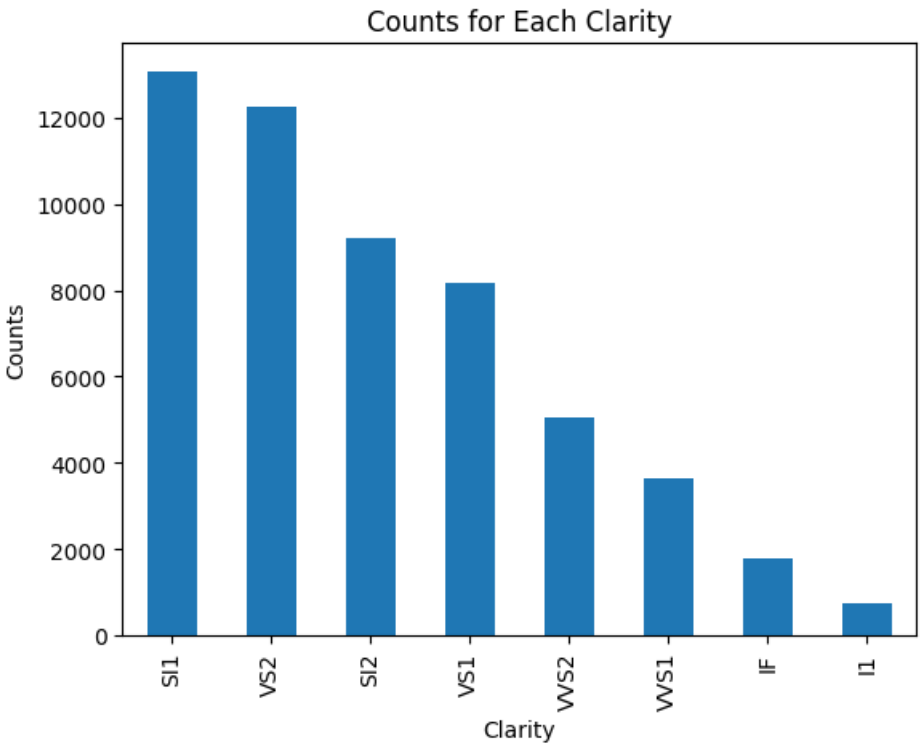
In [12]:
```
diamonds_original['cut'].value_counts().plot(kind='bar')
plt.xlabel('Cut')
plt.ylabel('Counts')
plt.title('Counts for Each Cut')
```

Out[12]: Text(0.5, 1.0, 'Counts for Each Cut')



In [13]:
```
diamonds_original['clarity'].value_counts().plot(kind='bar')
plt.xlabel('Clarity')
plt.ylabel('Counts')
plt.title('Counts for Each Clarity')
```

Out[13]: Text(0.5, 1.0, 'Counts for Each Clarity')



Question 2.1

Standardize feature columns and prepare them for training.

From Piazza:

- Split the dataset, and do fit_transform() on training set and transform() on test set.
- Only standardize train set.
- No, the standardization is not needed on the target variable. Yes, the RMSE will be larger.
  https://piazza.com/class/lcjwqyyip6bid/post/220 (https://piazza.com/class/lcjwqyyip6bid/post/220)
  https://piazza.com/class/lcjwqyyip6bid/post/223 (https://piazza.com/class/lcjwqyyip6bid/post/223)

In [114]:
```
Xdiamond = diamonds_data.loc[:, diamonds_data.columns != 'price']
Ydiamond = diamonds_data.price
X_train, X_test, y_train, y_test = train_test_split(Xdiamond, Ydiamond, test_size=0.1, ran
y_train = y_train.to_frame()
```

In [115]:
```
scaler = StandardScaler()
X_train_standard = scaler.fit_transform(X_train)
X_test_standard = scaler.transform(X_test)
```

Question 2.2

Describe how Feature Selection qualitatively affects the performance of your models in terms of test RMSE. Is it true for all model types? Also list two features for either dataset that has the lowest MI w.r.t to the target.

Ans: Mutual information tells us the dependency between two variables, while F score tells us whether an added new variable will improve a model. With these scores, we can avoid taking the less relevant features into consideration and prevent overfitting. From the plot below, we can see that for the three models, the decrase of RMSE halts at a certain point (k=6), since the most relevant features have already been selected. For the diamonds dataset, the two features with the lowest MI w.r.t. price are: table (MI=0.036565) and depth (MI=0.031448).

In [116]:

```
RMSE_MIR_OLS, RMSE_F_OLS = [], []
RMSE_MIR_Lasso, RMSE_F_Lasso = [], []
RMSE_MIR_RR, RMSE_F_RR = [], []

for i in range(X_train_standard.shape[1]):
    print('k = ', i+1)
    Xdiamond_MIR = SelectKBest(score_func=mutual_info_regression, k=i+1).fit_transform(X_t
    Xdiamond_F = SelectKBest(score_func=f_regression, k=i+1).fit_transform(X_train_standar

    print('Ordinary Least Squares')
    res1 = cross_validate(LinearRegression(), Xdiamond_MIR, y_train.values.ravel(), scorir
    RMSE_MIR_OLS.append(-(res1['test_neg_root_mean_squared_error'].mean()))
    res2 = cross_validate(LinearRegression(), Xdiamond_F, y_train.values.ravel(), scoring=
    RMSE_F_OLS.append(-(res2['test_neg_root_mean_squared_error'].mean()))
    print('Average RMSE Error for Mutual Info Regression:', RMSE_MIR_OLS[-1])
    print('Average RMSE Error for F Regression:', RMSE_F_OLS[-1])

    print('Lasso')
    res1 = cross_validate(Lasso(), Xdiamond_MIR, y_train.values.ravel(), scoring=['neg_roc
    RMSE_MIR_Lasso.append(-(res1['test_neg_root_mean_squared_error'].mean()))
    res2 = cross_validate(Lasso(), Xdiamond_F, y_train.values.ravel(), scoring=['neg_root_
    RMSE_F_Lasso.append(-(res2['test_neg_root_mean_squared_error'].mean()))
    print('Average RMSE Error for Mutual Info Regression:', RMSE_MIR_Lasso[-1])
    print('Average RMSE Error for F Regression:', RMSE_F_Lasso[-1])

    print('Ridge Regression')
    res1 = cross_validate(Ridge(), Xdiamond_MIR, y_train.values.ravel(), scoring=['neg_roc
    RMSE_MIR_RR.append(-(res1['test_neg_root_mean_squared_error'].mean()))
    res2 = cross_validate(Ridge(), Xdiamond_F, y_train.values.ravel(), scoring=['neg_root_
    RMSE_F_RR.append(-(res2['test_neg_root_mean_squared_error'].mean()))
    print('Average RMSE Error for Mutual Info Regression:', RMSE_MIR_RR[-1])
    print('Average RMSE Error for F Regression:', RMSE_F_RR[-1])
```

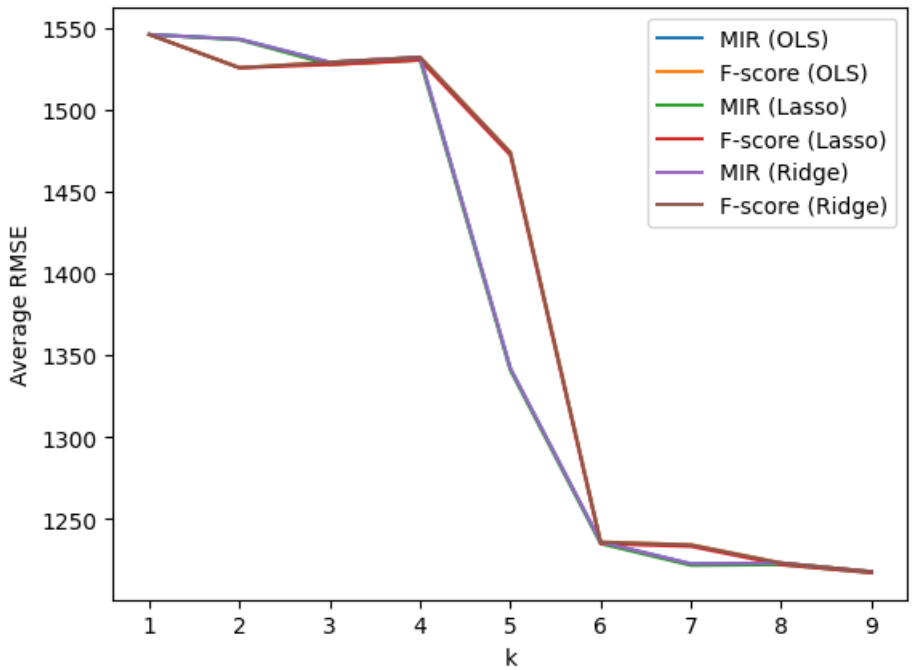
```
k =  1
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1545.8955641911862
Average RMSE Error for F Regression: 1545.8955641911862
Lasso
Average RMSE Error for Mutual Info Regression: 1545.89599426108
Average RMSE Error for F Regression: 1545.89599426108
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1545.8955726699164
Average RMSE Error for F Regression: 1545.8955726699164
k =  2
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1543.0423357852374
Average RMSE Error for F Regression: 1525.5664194317942
Lasso
Average RMSE Error for Mutual Info Regression: 1542.6964649319766
Average RMSE Error for F Regression: 1525.5947010354823
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1543.0206431037318
Average RMSE Error for F Regression: 1525.5663725550007
k =  3
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1528.9467895431847
Average RMSE Error for F Regression: 1528.9467895431847
Lasso
Average RMSE Error for Mutual Info Regression: 1527.5060350081812
Average RMSE Error for F Regression: 1527.5060350081812
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1528.9077098133243
Average RMSE Error for F Regression: 1528.9077098133243
k =  4
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1532.0262643129868
Average RMSE Error for F Regression: 1532.0262643129868
Lasso
Average RMSE Error for Mutual Info Regression: 1530.276596137666
Average RMSE Error for F Regression: 1530.276596137666
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1531.975635742033
Average RMSE Error for F Regression: 1531.975635742033
k =  5
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1341.8625201499462
Average RMSE Error for F Regression: 1473.928421755844
Lasso
Average RMSE Error for Mutual Info Regression: 1340.7301011772279
Average RMSE Error for F Regression: 1472.1950159461378
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1341.8373759270326
Average RMSE Error for F Regression: 1473.8782974260325
k =  6
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1235.8311097381393
Average RMSE Error for F Regression: 1235.8311097381393
Lasso
Average RMSE Error for Mutual Info Regression: 1234.8266266013304
Average RMSE Error for F Regression: 1234.8266266013304
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1235.8095254235318
Average RMSE Error for F Regression: 1235.8095254235318
k =  7
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1222.53138190742
Average RMSE Error for F Regression: 1234.1744956888426
Lasso
Average RMSE Error for Mutual Info Regression: 1221.586292167437
Average RMSE Error for F Regression: 1233.1910577495944
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1222.513229191995
Average RMSE Error for F Regression: 1234.1491117596986
k =  8
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1223.0389431055728
Average RMSE Error for F Regression: 1223.0389431055728
Lasso
Average RMSE Error for Mutual Info Regression: 1222.0614057886435
```

```
Average RMSE Error for F Regression: 1222.0614057886435
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1223.0193229643892
Average RMSE Error for F Regression: 1223.0193229643892
k =  9
Ordinary Least Squares
Average RMSE Error for Mutual Info Regression: 1217.7263398590414
Average RMSE Error for F Regression: 1217.7263398590414
Lasso
Average RMSE Error for Mutual Info Regression: 1217.0906429027702
Average RMSE Error for F Regression: 1217.0906429027702
Ridge Regression
Average RMSE Error for Mutual Info Regression: 1217.7136613826844
Average RMSE Error for F Regression: 1217.7136613826844
```

In [117]:

```
plt.plot(np.arange(1, 10), RMSE_MIR_OLS)
plt.plot(np.arange(1, 10), RMSE_F_OLS)
plt.plot(np.arange(1, 10), RMSE_MIR_Lasso)
plt.plot(np.arange(1, 10), RMSE_F_Lasso)
plt.plot(np.arange(1, 10), RMSE_MIR_RR)
plt.plot(np.arange(1, 10), RMSE_F_RR)
plt.legend(['MIR (OLS)','F-score (OLS)', 'MIR (Lasso)', 'F-score (Lasso)', 'MIR (Ridge)',
plt.xlabel('k')
plt.ylabel('Average RMSE')
```

Out[117]: Text(0, 0.5, 'Average RMSE')



Choose k=6.

```
mutual_info = mutual_info_regression(Xdiamond, Ydiamond)
mutual_info = pd.Series(mutual_info)
mutual_info.index = Xdiamond.columns
mutual_info.sort_values(ascending=False)

#Xdiamond_select = diamonds_data.loc[:, ['carat', 'x', 'y', 'z', 'clarity_numeric',
'color_numeric', 'cut_numeric']]
```

3.3 Evaluation

Perform 10-fold cross-validation and measure average RMSE errors for training and validation sets.

Question 3

For random forest model, measure “Out-of-Bag Error” (OOB) as well. Explain what OOB error and R2 score means given this link.

From Piazza 217 https://piazza.com/class/lcjwqyyjp6bid/post/217 (https://piazza.com/class/lcjwqyyjp6bid/post/217)
Q3: asks you to define it generally.
Q7.4: asks you to evaluate the values with respect to the model trained in this section.
OOB error measures the prediction error of random forests, boosted decision trees etc. with bootstrap aggregating, which uses subsampling with replacement to create training samples. OOB error is the mean prediction error on each training sample. Since the calculation only uses trees that did not have the training sample in the bootstrap sample, it serves as a form of validation score for the ensemble.
R2 is the ratio of the variance in the target variable that can be predicted by the features in the dataset. It includes all data in the training and validation set, not considering whether a sample is within a tree or not.

Reference: https://en.wikipedia.org/wiki/Out-of-bag_error#:~:text=Out%2Dof%2Dbag%20 (https://en.wikipedia.org/wiki/Out-of-bag_error#:~:text=Out%2Dof%2Dbag%20)
(OOB,the%20model%20to%20learn%20from.

What is the objective function? Train three models: (a) ordinary least squares (linear regression without regularization), (b) Lasso and (c) Ridge regression, and answer the following questions:

Question 4.1

Explain how each regularization scheme affects the learned parameter set.

Ans:
Lasso regression is linear regression with L1 regularization, which tends to shrink some parameter values to zero, and can result in sparse weights. It prevents overfitting and encourages feature selection. When the penalty term is weak, Lasso regression will be similar to the original linear regression, and will only make little changes to the learned parameter set compared to simply using linear regression. As the penalty term increases, more parameter values will become zero and result in a sparse parameter set. However, if the penalty term is too strong and turns most of the parameters to zero, we are no longer extracting useful features with the model.
As for ridge regression, it is linear regression with L2 regularization, which also shrinks weights, but does not reduce any parameter to zero, since L2 assumes the priors on weights are sampled from a unit Gaussian distribution. It helps reduce collinearity effects of the features. The penalty term tends to produce diffuse weights, and stabilize the model by promoting participation of all the features.

Question 4.2

Report your choice of the best regularization scheme along with the optimal penalty parameter and explain how you computed it.

To compute the optimal penalty parameter, we performed grid search with 10-fold cross validation, set λ in the range of [10^−5, 10^5], and applied on models 'Lasso' and 'Ridge' respectively. In addition, the differences between standardized and non-standardized datasets were also tested using X_train and X_train_standard.
We chose the best regularization scheme to be: Lasso, F-score feature selection, alpha=1, which results in Test RMSE= 1234.7313838620905.

In [118]:

```
k = 6
Xdiamond_MIR = SelectKBest(score_func=mutual_info_regression, k=k).fit_transform(X_train,
Xdiamond_MIRS = SelectKBest(score_func=mutual_info_regression, k=k).fit_transform(X_train_
Xdiamond_F = SelectKBest(score_func=f_regression, k=k).fit_transform(X_train, y_train.valu
Xdiamond_FS = SelectKBest(score_func=f_regression, k=k).fit_transform(X_train_standard, y_
```

Ordinary Least Squares

In [119]:

```
print('No Standardizaion')
res1 = cross_validate(LinearRegression(), Xdiamond_MIR, y_train, scoring=['neg_root_mean_s
print('MIR: Train RMSE=', -(res1['train_neg_root_mean_squared_error'].mean()), ', Test RMS
res2 = cross_validate(LinearRegression(), Xdiamond_F, y_train, scoring=['neg_root_mean_squ
print('F-Score: Train RMSE=', -(res2['train_neg_root_mean_squared_error'].mean()), ', Test
print('Standardizaion')
res1 = cross_validate(LinearRegression(), Xdiamond_MIRS, y_train, scoring=['neg_root_mean_
print('MIR: Train RMSE=', -(res1['train_neg_root_mean_squared_error'].mean()), ', Test RMS
res2 = cross_validate(LinearRegression(), Xdiamond_FS, y_train, scoring=['neg_root_mean_sc
print('F-Score: Train RMSE=', -(res2['train_neg_root_mean_squared_error'].mean()), ', Test
```

```
No Standardizaion
MIR: Train RMSE= 1231.950403808161 , Test RMSE= 1235.8311097381393
F-Score: Train RMSE= 1231.950403808161 , Test RMSE= 1235.8311097381393
Standardizaion
MIR: Train RMSE= 1231.950403808161 , Test RMSE= 1235.8311097381393
F-Score: Train RMSE= 1231.950403808161 , Test RMSE= 1235.8311097381393
```

Lasso

In [127]:

```
def search_penalty(model, params):
    grid_mir = GridSearchCV(model, params, scoring='neg_root_mean_squared_error', n_jobs=-
    grid_mir_s = GridSearchCV(model, params, scoring='neg_root_mean_squared_error', n_jobs
    grid_f = GridSearchCV(model, params, scoring='neg_root_mean_squared_error', n_jobs=-1,
    grid_f_s = GridSearchCV(model, params, scoring='neg_root_mean_squared_error', n_jobs=-
    print('No standardization')
    print('MIR: Train RMSE=', -(max(grid_mir.cv_results_['mean_train_score'])), '; Test RM
    print('F-score: Train RMSE=', -(max(grid_f.cv_results_['mean_train_score'])), '; Test
    print('Standardiazation')
    print('MIR: Train RMSE=', -(max(grid_mir_s.cv_results_['mean_train_score'])), '; Test
    print('F-score: Train RMSE=', -(max(grid_f_s.cv_results_['mean_train_score'])), '; Tes
```

In [128]:

```
pipe_lasso = Pipeline([('model', Lasso(random_state=77))])
param_grid = {'model__alpha': [10.0**x for x in np.arange(-5,5)]}
search_penalty(pipe_lasso, param_grid)
```

```
No standardization
MIR: Train RMSE= 1231.9504038081723 ; Test RMSE= 1234.7313838620905 ; alpha= 1.0
F-score: Train RMSE= 1231.9504038081723 ; Test RMSE= 1234.7313838620905 ; alpha= 1.0
Standardiazation
MIR: Train RMSE= 1231.9504038081677 ; Test RMSE= 1234.8266266013304 ; alpha= 1.0
F-score: Train RMSE= 1231.9504038081677 ; Test RMSE= 1234.8266266013304 ; alpha= 1.0
```

Ridge Regression

In [129]:

```
pipe_ridge = Pipeline([('model', Ridge(random_state=77))])
param_grid = {'model__alpha': [10.0**x for x in np.arange(-5,5)]}
search_penalty(pipe_ridge, param_grid)
```

```
No standardization
MIR: Train RMSE= 1231.950403808161 ; Test RMSE= 1235.6777883853667 ; alpha= 10.0
F-score: Train RMSE= 1231.950403808161 ; Test RMSE= 1235.6777883853667 ; alpha= 10.0
Standardiazation
MIR: Train RMSE= 1231.950403808161 ; Test RMSE= 1235.548907697946 ; alpha= 100.0
F-score: Train RMSE= 1231.950403808161 ; Test RMSE= 1235.548907697946 ; alpha= 100.0
```

Question 4.3

Does feature standardization play a role in improving the model performance (in the cases with ridge regularization)? Justify your answer.

> As we can see in the results above, there is no too much difference in the performance between model with standardization or not.

Question 4.4

Some linear regression packages return p-values for different features. What is the meaning of these p-values and how can you infer the most significant features?

> p-value measures the probability of obtaining an observed result under the null hypothesis. The p-value for each feature assumes the null hypothesis "a feature has no correlation with the target variable" is true. The lower the p-value, the more likely we are to reject the null hypothesis. In the results below, we see some p-values close to 0. For the corresponding features, we have more confidence to say that this feature does not have no correlation with the target variable. The mose significant features are 'carat', 'color_numeric', and 'clarity_numeric'.

In [138]:

```
lr = OLS(y_train, X_train).fit()
print(lr.pvalues.sort_values())
```

```
carat               0.000000e+00
color_numeric       0.000000e+00
clarity_numeric     0.000000e+00
cut_numeric        4.101860e-148
depth              4.804137e-129
x                  3.657069e-116
table               1.690737e-10
z                   1.584959e-02
y                   1.880248e-02
dtype: float64
```

Question 5.1

What are the most salient features? Why?

> Below we performed grid search and figured that 'carat' and the diemnsions 'x', 'y', and 'z' are the most salient features. The top terms are composed of these features and their combinations.

In [139]:

```
pipe_poly = Pipeline([('PR', PolynomialFeatures()), ('model', Ridge(random_state=77))])
param_poly = {'PR__degree': np.arange(2, 7), #Piazza 224: min degree = 2
                 'model__alpha': [10.0**x for x in np.arange(-5,5)]}
grid_poly = GridSearchCV(pipe_poly, param_grid=param_poly, cv=10, n_jobs=-1, verbose=False
```

```
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/linear_model/_ridg
e.py:216: LinAlgWarning: Ill-conditioned matrix (rcond=9.70494e-17): result may not be accur
ate.
  return linalg.solve(A, Xy, assume_a="pos", overwrite_a=True).T
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/linear_model/_ridg
e.py:216: LinAlgWarning: Ill-conditioned matrix (rcond=7.03373e-19): result may not be accur
ate.
  return linalg.solve(A, Xy, assume_a="pos", overwrite_a=True).T
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/linear_model/_ridg
e.py:216: LinAlgWarning: Ill-conditioned matrix (rcond=1.3529e-19): result may not be accura
te.
  return linalg.solve(A, Xy, assume_a="pos", overwrite_a=True).T
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/linear_model/_ridg
e.py:216: LinAlgWarning: Ill-conditioned matrix (rcond=3.60136e-19): result may not be accur
ate.
  return linalg.solve(A, Xy, assume_a="pos", overwrite_a=True).T
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/linear_model/_ridg
e.py:216: LinAlgWarning: Ill-conditioned matrix (rcond=4.69571e-19): result may not be accur
ate.
  return linalg.solve(A, Xy, assume_a="pos", overwrite_a=True).T
```

In [147]:

```
X_select = SelectKBest(score_func=f_regression, k=6)
X_fit = X_select.fit_transform(X_train, y_train.values.ravel())
features = X_train.columns[X_select.get_support()]
params = grid_poly.best_estimator_.get_params()
coefs = params['model'].coef_
feature_names = params['PR'].get_feature_names_out(list(features))
idx = np.argsort(-abs(coefs))
salient_features = [feature_names[i] for i in idx]
print ('Top Salient Features:', salient_features[0][:10])
```

```
Top Salient Features: ['carat clarity_numeric' 'carat color_numeric' 'carat y' 'carat x'
 'carat z' 'x' 'y' 'x z' 'z' 'carat^2']
```

Question 5.2

What degree of polynomial is best? How did you find the optimal degree? Whatdoes a very high-order polynomial imply about the fit on the training data? What about its performance on testing data?

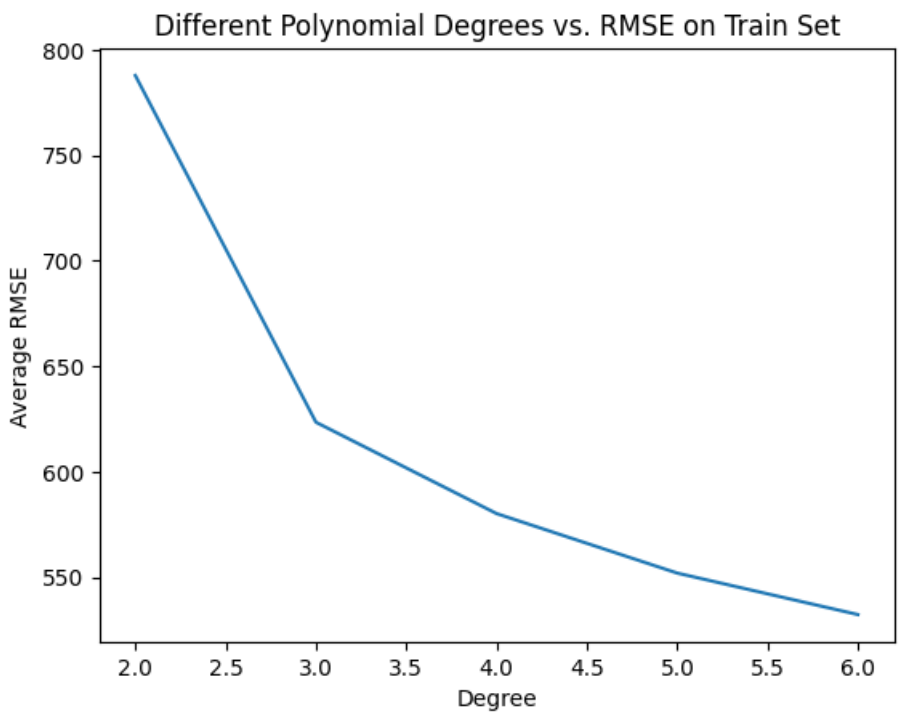
> Choose degree=5. We performed grid search on PolynomialFeatures, ranging the degrees from 2~6. We also plotted RMSE of the training and testing data over different degrees and saw that at degree=5, training error became desirably low but on the other hand, testing error is starting to rise. Therefore, we won't further increase the degree and get into the risk of overfitting.

In [148]:

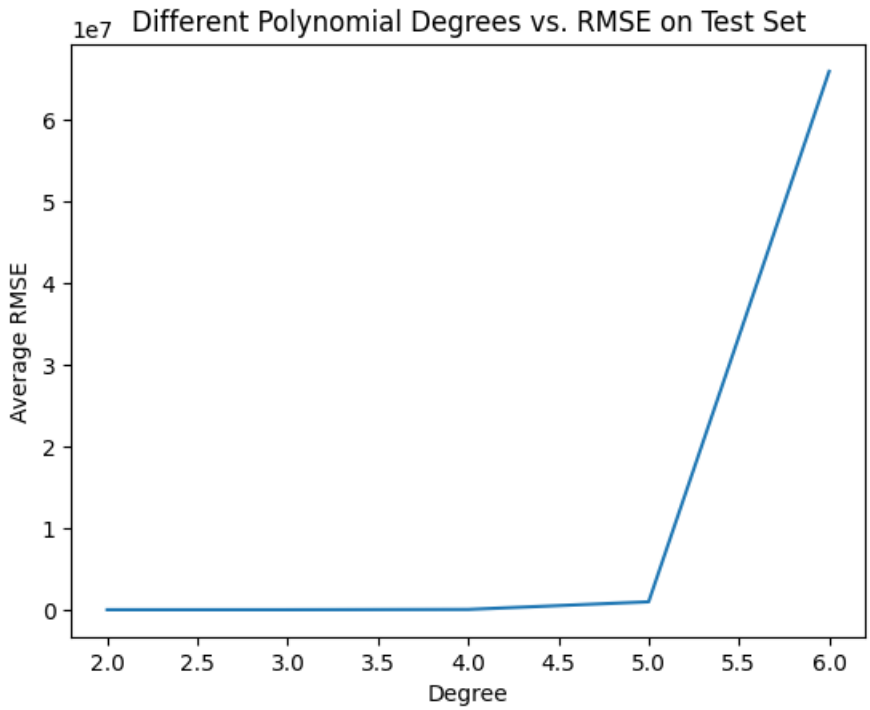
```
res = pd.DataFrame(grid_poly.cv_results_)[['mean_train_score', 'mean_test_score', 'param_P
train, test, alpha = [], [], []
for i in np.arange(2, 7):
    train.append(-(res.loc[res['param_PR__degree'] == i]).max().mean_train_score)
    test.append(-(res.loc[res['param_PR__degree'] == i]).max().mean_test_score)
```

In [149]:

```
plt.plot(np.arange(2, 7), train)
plt.xlabel('Degree')
plt.ylabel('Average RMSE')
plt.title('Different Polynomial Degrees vs. RMSE on Train Set')
plt.show()
```



In [150]:
```
plt.plot(np.arange(2, 7), test)
plt.xlabel('Degree')
plt.ylabel('Average RMSE')
plt.title('Different Polynomial Degrees vs. RMSE on Test Set')
plt.show()
```



Question 6.1

Adjust your network size (number of hidden neurons and depth), and weight decay as regularization. Find a good hyper-parameter set systematically (no more than 20 experiments in total).
(Side note:

- Piazza posts concerning "weight decay" #248, #258 https://piazza.com/class/lcjwqyyjp6bid/post/258 (https://piazza.com/class/lcjwqyyjp6bid/post/258) don't really answer the question. Here I'm assuming weight decay means beta_1 & 2 and didn't set anything additionally. https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html (https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html)
- "no more than 20 experiments" I'm assuming it refers to network sizes.)

> Ans: We tested network sizes of depths 1~3, and each layer may contain 10, 20, 30 neurons respectively. Along with different activation functions, 'logistic', 'tanh', 'relu', and alphas from np.logspace(-1, 1, 5), we used GridSearchCV to search for the best parameter combinations. The results showed that the best combination is {network size: (30, 30, 30), activation function: 'relu', alpha: 3.1622776601683795}, with train RMSE: 602.1248387911928 and test RMSE: 614.2593043390436.

In [151]:
```
neurons = [10, 20, 30]
combinations = []
for i in range(len(neurons)+1):
    items = list(itertools.combinations_with_replacement(neurons, i))
    combinations += items
combinations = combinations[1:]
```

In [176]:
```
print('All network structures:', combinations)
```

```
All network structures: [(10,), (20,), (30,), (10, 10), (10, 20), (10, 30), (20, 20), (20, 3
0), (30, 30), (10, 10, 10), (10, 10, 20), (10, 10, 30), (10, 20, 20), (10, 20, 30), (10, 30,
30), (20, 20, 20), (20, 20, 30), (20, 30, 30), (30, 30, 30)]
```

In [152]:

```
pipe_nn = Pipeline([('model', MLPRegressor(random_state=77, max_iter=1000))]) #default sol
param_nn = {'model__hidden_layer_sizes': combinations,
            'model__activation': ['logistic', 'tanh', 'relu'],
            'model__alpha': np.logspace(-1, 1, 5)}
grid_nn = GridSearchCV(pipe_nn, param_grid=param_nn, cv=10, n_jobs=-1, verbose=False, scor
```

```
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/neural_network/_mu
ltilayer_perceptron.py:1593: DataConversionWarning: A column-vector y was passed when a 1d a
rray was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/neural_network/_mu
ltilayer_perceptron.py:1593: DataConversionWarning: A column-vector y was passed when a 1d a
rray was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/neural_network/_mu
ltilayer_perceptron.py:1593: DataConversionWarning: A column-vector y was passed when a 1d a
rray was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/neural_network/_mu
ltilayer_perceptron.py:1593: DataConversionWarning: A column-vector y was passed when a 1d a
rray was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/neural_network/_mu
ltilayer_perceptron.py:1593: DataConversionWarning: A column-vector y was passed when a 1d a
rray was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

In [154]:

```
res_nn = pd.DataFrame(grid_nn.cv_results_)[['mean_train_score', 'mean_test_score', 'param_
print('Best parameters:', grid_nn.best_params_, 'Train RMSE:', -(max(res_nn.mean_train_sco
```

```
Best parameters: {'model__activation': 'relu', 'model__alpha': 3.1622776601683795, 'model__h
idden_layer_sizes': (30, 30, 30)} Train RMSE: 602.1248387911928 , Test RMSE: 614.25930433904
36
```

Question 6.2

How does the performance generally compare with linear regression? Why?

> Ans:
> The table below shows the comparison between the best results of linear regression (with L1 and L2 regression) and neural network, and we can see that neural network performs significantly better, reaching training and testing RMSE at around 600, which is half of that produced by linear regression. This is because compared to linear regression, neural networks capture the non-linearity relationship between the input and the target variable. In addition, with the hidden layers and neurons, neural networks are able handle high-dimensional data and capture only the relevant features.

Method	Linear Regression	Lasso	Ridge	Neural Network
Train RMSE	1231.95	1231.95	1231.95	602.12
Test RMSE	1235.83	1234.83	1235.55	614.26

Question 6.3

What activation function did you use for the output and why? You may use none.

> Ans: Of all the combinations, the network with 'relu' activation gives the best results. ReLU is a linear function when the input is >0, and is often used for regression or classification problems, especially when the target variable is non-negative. Our task is a regression problem aiming to predict the price, where ReLU perfectly suits. In addition, ReLU has the advantages of computational efficiency and is less likely to suffer from vanishing gradients.

Question 6.4

What is the risk of increasing the depth of the network too far?

Ans:

1. Vanishing or exploding gradients: Since a great part of neural networks relies on back-propagation, we should consider that as we deepen a network, the gradients will become exponentially small or large when we reach back to the earlier layers, and may lead to convergence to a sub-optimal point.
2. Overfitting: Deeper neural networks will inevitably be more complex and have more trainable parameters, which might overfit the training data and could not generalize well on the test data.
3. Memory and time intensive: A deeper network takes more time to train and also needs more memory to store the variables.
4. Interpretability: Large amount of parameters will lead to complex relationships between features and target variables, and can make it hard to interpret intuitively.

Question 7.1

Explain how these hyper-parameters affect the overall performance. Describe if and how each hyper-parameter results in a regularization effect during training.

Ans:

1. Maximum number of features: From our results, we see that the training error decreased steadily as the the number of features increased, but for the testing error, the decrease halted at a certain point. Since increasing the number of feature improves model capacity for each tree, but also increrases the correlation between different trees and will eventually lead to overfitting. Only selecting the important features can help with better generalization and prevent overfitting.
2. Number of trees: Since each tree results from a set of randomized individual bootstrap subsample, the growth of different trees does not affect one another. Increasing the number of trees itself generally does not cause overfitting, but can act as a regularization effect and lead to better performance. However, it may increases computational and training complexity.
3. Depth of each tree: As shown in the plot below, we see that the error rate decreased significantly for the training dataset as we increased the tree depth. However, for the testing dataset, after reaching a minimal error at depth=10, the error rate starts to once again increase slightly. Since increasing the tree depth also increses the number of splits, it will eventually cause overfitting by considering too many complex patterns. A shallower tree may have a regularization effect by promoting generalization and preventing overfitting.

In [155]:

```
pipe_rf = Pipeline([('model', RandomForestRegressor(random_state=77, oob_score=True))])
param_rf = {'model__max_features': np.arange(1, 5),
            'model__n_estimators': np.arange(10, 210, 10),
            'model__max_depth': np.arange(1, 21)}
grid_rf = GridSearchCV(pipe_rf, param_grid=param_rf, cv=10, n_jobs=-1, verbose=False, scor
res_rf = pd.DataFrame(grid_rf.cv_results_)[['mean_train_score', 'mean_test_score', 'param_
print('Best parameters:', grid_rf.best_params_, 'Train RMSE:', -(max(res_rf.mean_train_scc
```

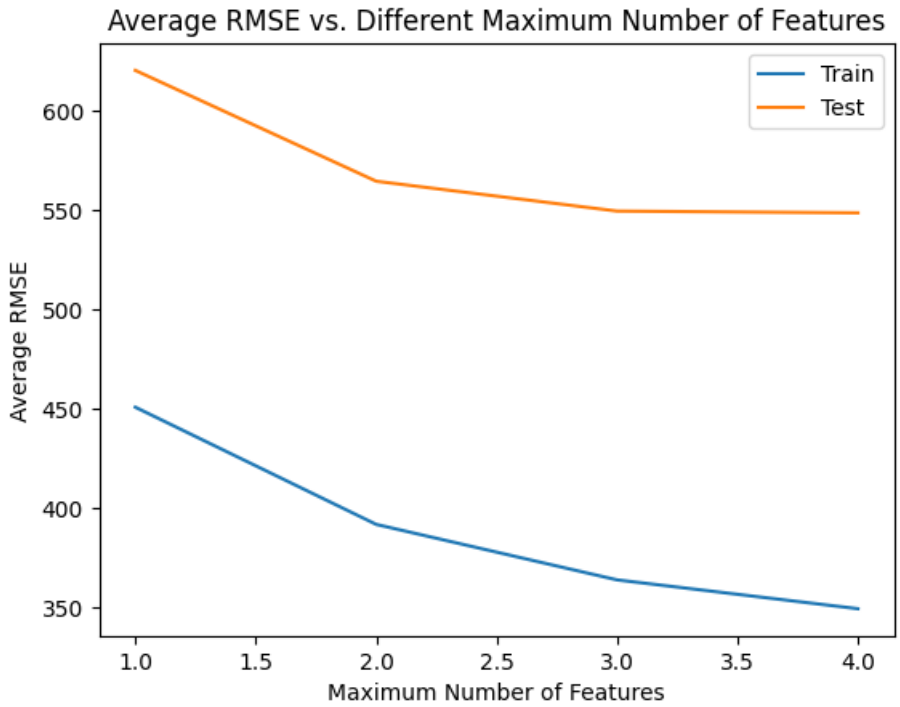
```
  self._final_estimator.fit(Xt, y, **fit_params_last_step)
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/pipeline.py:406: D
ataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chan
ge the shape of y to (n_samples,), for example using ravel().
  self._final_estimator.fit(Xt, y, **fit_params_last_step)
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/pipeline.py:406: D
ataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chan
ge the shape of y to (n_samples,), for example using ravel().
  self._final_estimator.fit(Xt, y, **fit_params_last_step)
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/pipeline.py:406: D
ataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chan
ge the shape of y to (n_samples,), for example using ravel().
  self._final_estimator.fit(Xt, y, **fit_params_last_step)
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-packages/sklearn/pipeline.py:406: D
ataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chan
ge the shape of y to (n_samples,), for example using ravel().
  self._final_estimator.fit(Xt, y, **fit_params_last_step)

Best parameters: {'model__max_depth': 14, 'model__max_features': 4, 'model__n_estimators': 1
30} Train RMSE: 220.10684767084336 , Test RMSE: 548.3859582686234
```

In [161]:

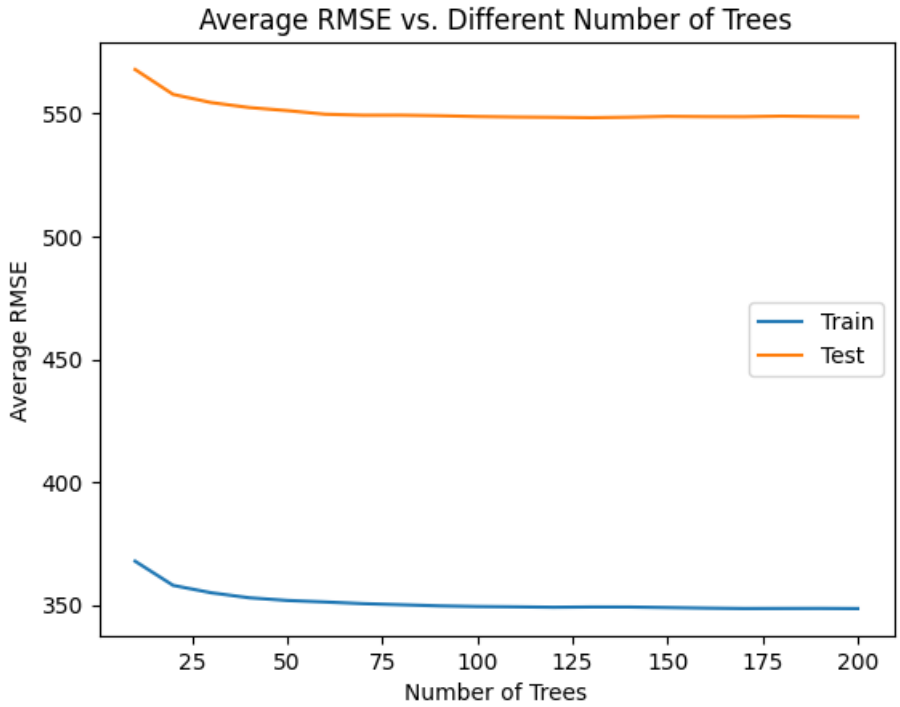
```
max_features = np.arange(1, 5).reshape(4)
n_estimators = np.arange(10, 210, 10).reshape(20)
max_depth = np.arange(1, 21).reshape(20)

def params_rmse(param1, n1, param2, n2):
    train = list(-((res_rf[(res_rf[param1] == n1) & (res_rf[param2] == n2)]).mean_train_sc
    test = list(-((res_rf[(res_rf[param1] == n1) & (res_rf[param2] == n2)]).mean_test_scor
    if (param1 != 'param_model__n_estimators') and (param2 != 'param_model__n_estimators')
        x_axis, x_param = n_estimators, 'Number of Trees'
    elif (param1 != 'param_model__max_depth') and (param2 != 'param_model__max_depth'):
        x_axis, x_param = max_depth, 'Depth of Each Tree'
    else: x_axis, x_param = max_features, 'Maximum Number of Features'
    plt.plot(x_axis, train)
    plt.plot(x_axis, test)
    plt.xlabel(x_param)
    plt.ylabel('Average RMSE')
    plt.title('Average RMSE vs. Different %s' %x_param)
    plt.legend(['Train', 'Test'])
    plt.show()
```

In [163]:

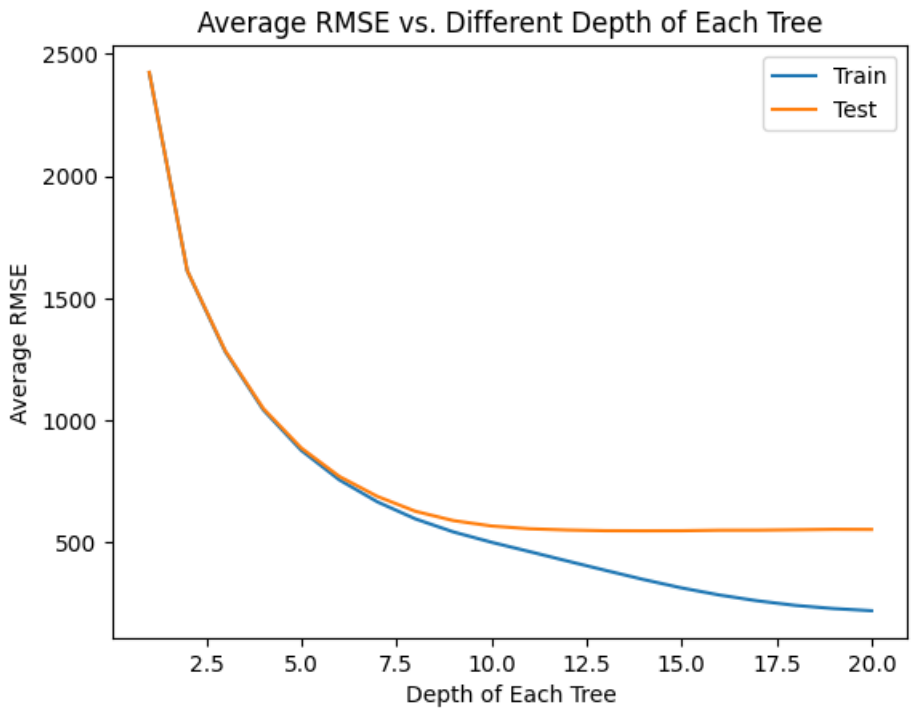
```
params_rmse('param_model__max_depth' , 14, 'param_model__n_estimators', 130)
```



In [162]:
```
params_rmse('param_model__max_depth', 14, 'param_model__max_features', 4)
```



In [164]:
```
params_rmse('param_model__max_features' , 4, 'param_model__n_estimators', 130)
```



Question 7.2

How do random forests create a highly non-linear decision boundary despite the fact that all we do at each layer is apply a threshold on a feature?

Ans: A random forest is an ensemble of individual decision trees. While for each layer we only simply apply a threshold on a feature, by combining different trees with different splits, features, and thresholds, we can get more sophisticated decision boundaries. Random forests combine multiple uncorrelated models and outperform each individual model because it captures the non-linearity relationships between features that may not show from a single decision tree.

Question 7.3

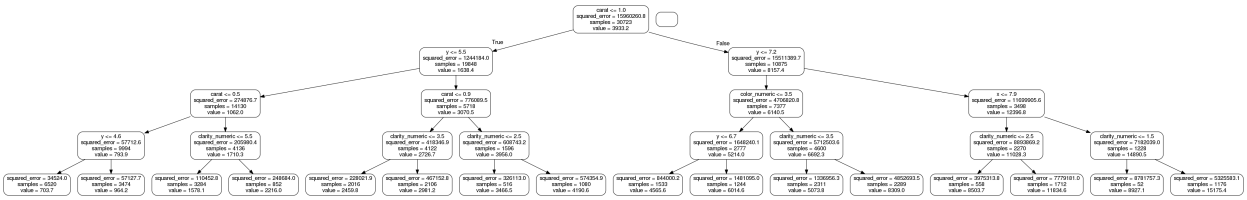
Randomly pick a tree in your random forest model (with maximum depth of 4) and plot its structure. Which feature is selected for branching at the root node? What can you infer about the importance of this feature as opposed to others? Do the important features correspond to what you got in part 3.3.1?

> Ans: The feature selected for branching at the root node is 'carat.' This makes sense because carat measures the weights for diamonds. According to GIA, larger diamonds are rarer and more desirable, therefore more pricey. Moreover, color, clarity, and cut are the three other important factors that affect the diamond price given that the carats are the same.
> Following 'carat,' we have 'y' and 'clarity_numeric' selected for branching. From 3.3.1, the top 5 salient features are carat, y, x, z, and clarity_numeric. While 'x' and 'z' are not selected for branching, they are a means of measuring diamond dimension as y is, and we assume they have similar effects when it comes to price prediction. Overall, the branching features are in line with the salient features selected previously.
> Reference: [https://4cs.gia.edu/en-us/diamond-carat-weight/ (https://4cs.gia.edu/en-us/diamond-carat-weight/)](https://4cs.gia.edu/en-us/diamond-carat-weight/)

In [166]:

```
rf = RandomForestRegressor(random_state=77, max_depth=4, max_features=4, n_estimators=130)
tree = rf.estimators_[1]
export_graphviz(tree, out_file = 'tree.dot', feature_names=features, rounded=True, precisi
(graph, ) = pydot.graph_from_dot_file('tree.dot')
Image(graph.create_png())
```

Out[166]:



Question 7.4

Measure “Out-of-Bag Error” (OOB). Explain what OOB error and R2 score means.

> Ans: The measured OOB score is 0.98. OOB score can be used if there is not enough data to split for train and validation sets. It resembles R2 score, but is calculated using only the training data by subsampling different portions for each calculation. The R2 score tells how successfully a model accounts for the intrinsic variation, and runs the trained model on the validation data. Both scores can be used to predict the generalizability of a model.

In [168]:

```
print('OOB error:', RandomForestRegressor(random_state=77, max_depth=19, max_features=2, r
```

```
OOB error: 0.9800773920038945
```

Question 8.1

Read the documentation of LightGBM OR CatBoost and determine the important hyperparameters along with a search space for the tuning of these parameters (keep the search space small).

LightGBM

The list of hyperparameters we consider important are as follows:

1. 'boosting_type': Selects the type of method for constructing a decision tree. Our search space includes: ‘gbdt’, traditional Gradient Boosting Decision Tree; ‘dart’, ‘rf’, Random Forest, and 'dart', Dropouts meet Multiple Additive Regression Trees.
2. 'num_tree': Number of boosting iterations. Our search space is: np.arange(10, 3000, 100)
3. 'num_leaves': Maximum tree leaves for base learners. Our search space is: np.arange(20, 1000, 20)
4. 'max_depth': Maximum tree depth for base learners. Our search space is: np.arange(1, 50, 5)
5. 'lambda_l1': Strength of L1 regularization. Our search space is: [10.0**x for x in np.arange(-4, 3)]
6. 'lambda_l2': Strength of L2 regularization. Our search space is: [10.0**x for x in np.arange(-4, 3)]
7. 'bagging': The fraction of the sampling, constrained to 0 < bagging <= 1.0. Our search space is: np.arange(0.1, 1, 0.1)
8. 'bagging_freq': The frequency for bagging. Our search space is: np.arange(0, 50, 5)
9. 'min_split_gain': The minimal gain to perform split, and can be used to speed up training. Our search space is: [10.0**x for x in np.arange(-4, 0)]
   Reference: https://lightgbm.readthedocs.io/en/latest/Parameters.html (https://lightgbm.readthedocs.io/en/latest/Parameters.html), https://lightgbm.readthedocs.io/en/latest/pythonapi/lightgbm.LGBMRegressor.html (https://lightgbm.readthedocs.io/en/latest/pythonapi/lightgbm.LGBMRegressor.html)

Question 8.2

Apply Bayesian optimization using skopt.BayesSearchCV from scikit-optmize to find the ideal hyperparameter combination in your search space. Report the best hyperparameter set found and the corresponding RMSE.

Ans: The best parameters are:

Parameter	Value
'bagging'	0.6
'bagging_freq'	10
'boosting_type'	'dart'
'lambda_l1'	1.0
'lambda_l2'	10.0
'max_depth'	41
'min_split_gain'	0.001
'num_leaves'	460
'num_tree'	510

rendering train RMSE: 195.34263528513097 and test RMSE: 546.5188445165489.

In [170]:

```
bayes_lgbm = BayesSearchCV(lgb.LGBMRegressor(random_state=77, verbose=1, n_jobs=-1),
                    {'boosting_type': ['gbdt', 'rf', 'dart'],
                     'num_tree': np.arange(10, 3000, 100),
                     'num_leaves': np.arange(20, 1000, 20),
                     'max_depth': np.arange(1, 50, 5),
                     'lambda_l1': [10.0**x for x in np.arange(-4, 3)],
                     'lambda_l2': [10.0**x for x in np.arange(-4, 3)],
                     'bagging': np.arange(0.1, 1, 0.1),
                     'bagging_freq': np.arange(0, 50, 5),
                     'min_split_gain': [10.0**x for x in np.arange(-4, 0)]},
                    n_iter=20, scoring='neg_root_mean_squared_error', n_jobs=-1, cv=10, ve
lgbm_fit = bayes_lgbm.fit(Xdiamond_F, y_train)
```

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

In [171]:

```
print('Best parameters:', bayes_lgbm.best_params_, 'Train RMSE:', -(max(bayes_lgbm.cv_resu
```

```
Best parameters: OrderedDict([('bagging', 0.6), ('bagging_freq', 10), ('boosting_type', 'dar
t'), ('lambda_l1', 1.0), ('lambda_l2', 10.0), ('max_depth', 41), ('min_split_gain', 0.001),
('num_leaves', 460), ('num_tree', 510)]) Train RMSE: 195.34263528513097 ; Test RMSE: 546.518
8445165489
```

Question 8.3

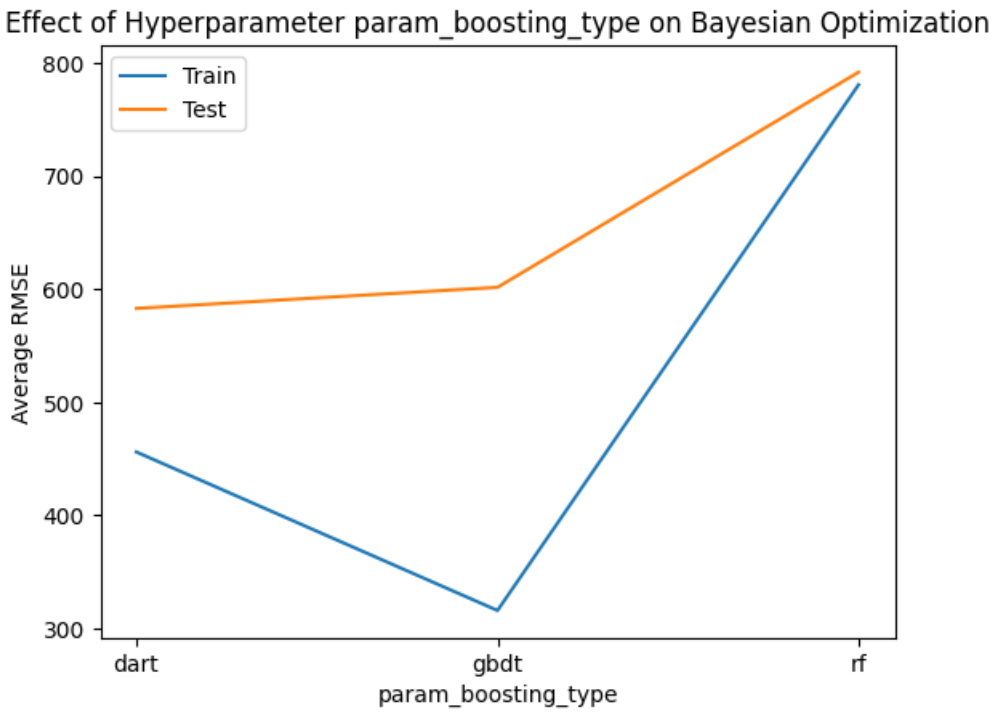
Qualitatively interpret the effect of the hyperparameters using the Bayesian optimization results: Which of them helps with performance? Which helps with regularization (shrinks the generalization gap)? Which affects the fitting efficiency?

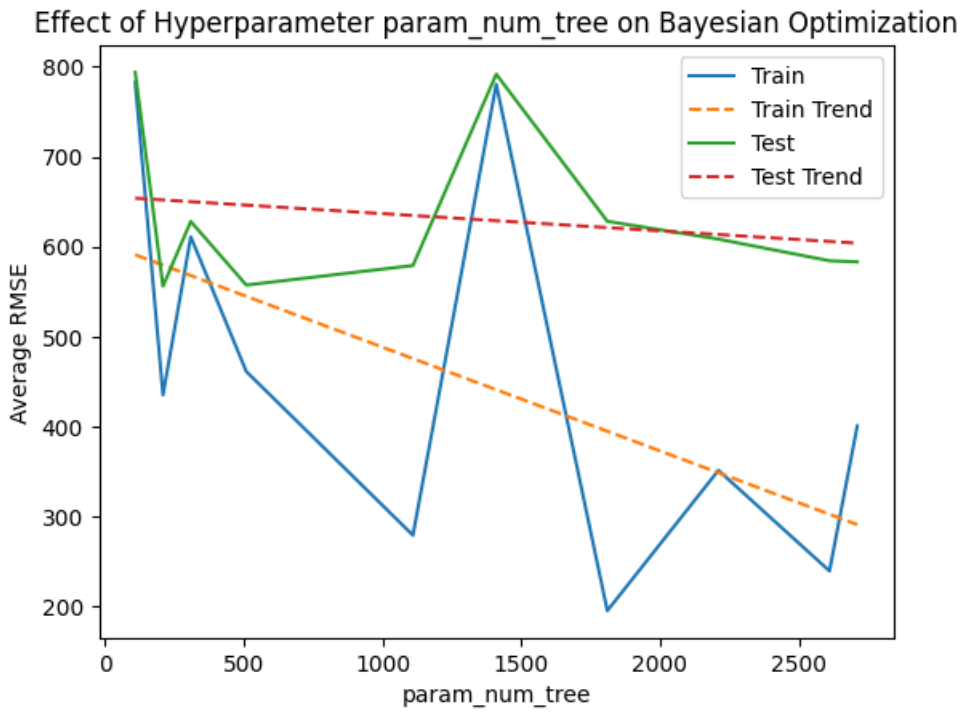
Ans: We plotted the error rate trend for each hyperparameter, and interpreted the results accordingly:

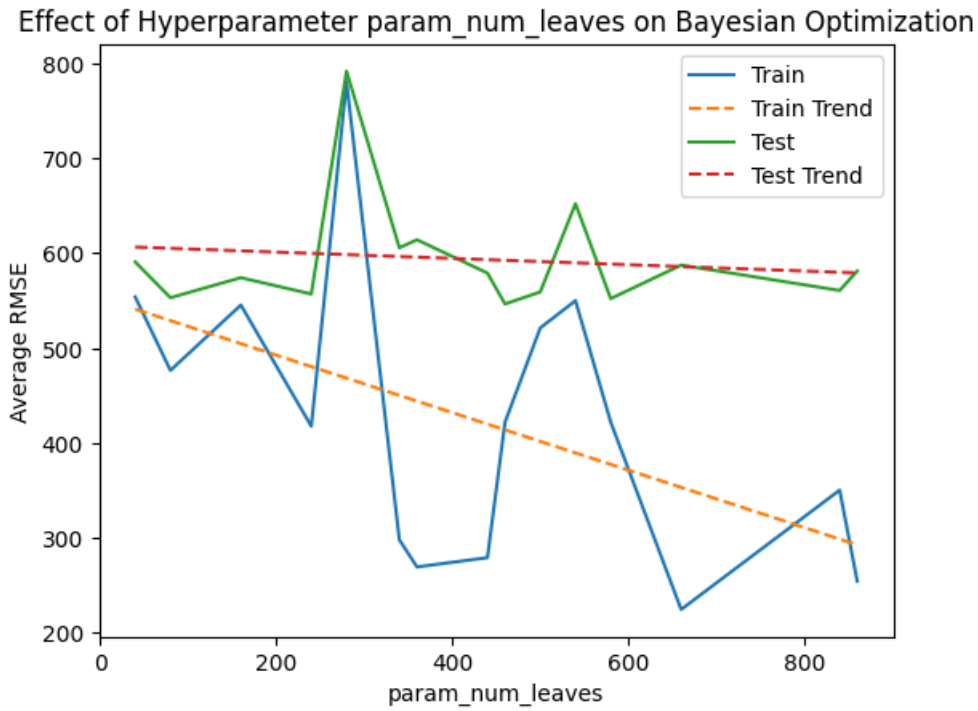
1. Boosting Type: gbdt is an ensemble model of decision trees, and is widely used because of its efficiency and stability. Each decision tree learns according to its previous tree. However, trees added at later iterations tend to make a negligible contribution towards the remaining instances. Therefore, 'dart' introduces dropout to make it more difficult for the trees at later iterations to specialize on only a few samples and hence improves the performance. While we expect 'dart' to perform better, our results did not align with the expectation. We assume it is because dart has other adjustable hyperparameters (e.g. probability of skipping dropout), which can make the results more unpredictable.
2. Number of Trees: As we increase the number of trees, we see the trend of decreasing RMSE. Since more trees do not cause overfitting, which is similar to random forests as mentioned above, increasing the number of trees may help stabilize the model and improve its performance.
3. Number of Leaves: Increasing the number of leaves helps decrease the error rate by taking more features into consideration. The hyperparameter acts as a regularizer and helps improve performance, but we have to avoid adding too many leaves to prevent the problem of overfitting.
4. Depth of Tree: Increasing the tree depth generally helps decrease the error rate. It acts as a regularizer and helps with performance as well. However, if the depth becomes too large, the model will also consider the effect of noise and cause overfitting. In conclusion, increasing tree depth decreases bias but increases variance.
5. L1: We can observe that both training and testing RMSE dropped significantly at the very beginning, but rose back up as we further increase the regularization strength. L1 aims to increase model generalization, but if the value is too high, the regularization term will dominate the original model when calculating the errors and become too noisy.
6. L2: Similar to L1 regularization, small regularization strength helps reduce RMSE, but we won't want the term to be too high to dominate the original model. We can also observe that L1 regularization is more robust than L2 as it may zero out some weights.
7. Bagging: As the subsampling ratio increases, the performance also imporves. Bagging helps reduce the variance of ensembling and has a regularizing effect when the subsampling ratio is small, even though very small values may cause models to perform poorly. On the other hand, large bagging ratio is similar to no bagging effect, which may cause the model to perform well on the training set but generalizes poorly on the testing set.
8. Bagging Frequency: As the frequency increases, there is a chance that RMSE rebounds. Since bagging acts as a regularizer, we can expect that the more frequent we perform bagging, the more likely we are to drop some important samples from the dataset and imperil the overall performance.
9. Minimal Split Gain: The plot indicates that as min_split_gain increases, there is minimal effect on decreasing RMSE. This hyperparameter mainly serves as an accelerator for speeding up fitting efficiency.
   Reference: [https://neptune.ai/blog/lightgbm-parameters-guide (https://neptune.ai/blog/lightgbm-parameters-guide)](https://neptune.ai/blog/lightgbm-parameters-guide)

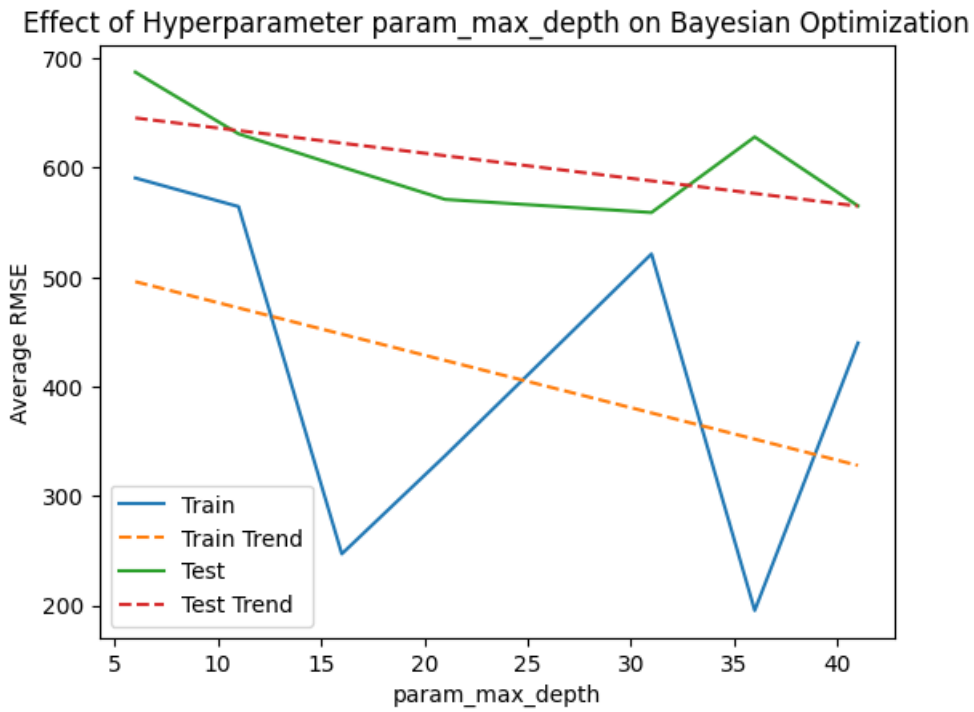
In [173]:

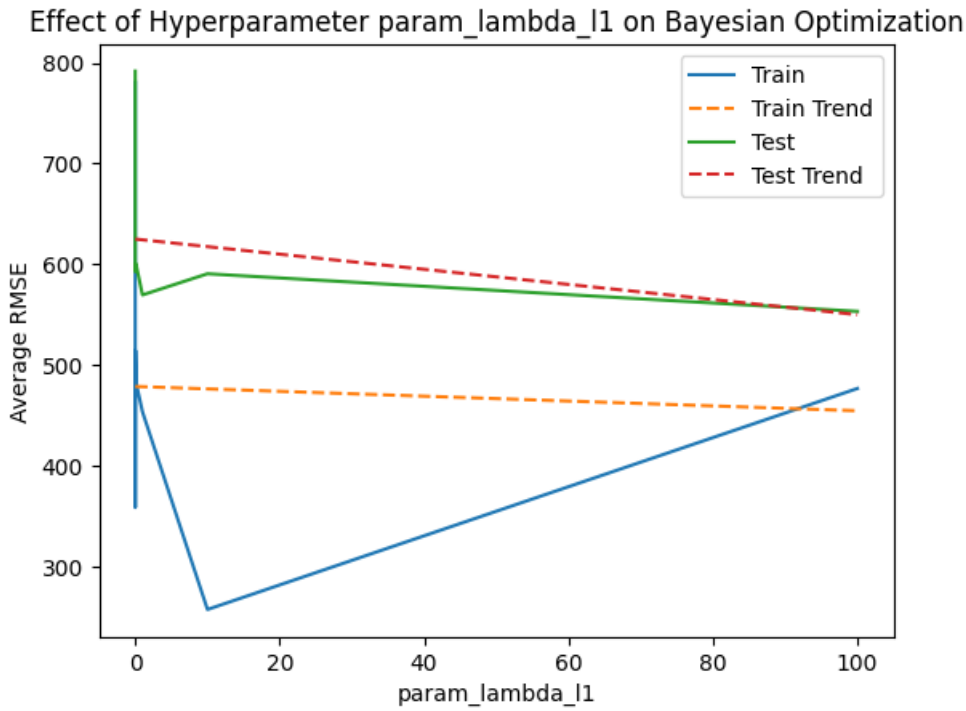
```
param_lgbm = ['param_boosting_type', 'param_num_tree', 'param_num_leaves', 'param_max_dept
              'param_lambda_l2', 'param_bagging', 'param_bagging_freq', 'param_min_split_g
for i in param_lgbm:
    param_set = sorted(list(set(bayes_lgbm.cv_results_[i])))
    param_train, param_test = [], []
    for j in param_set:
        param_train.append(-(np.mean([bayes_lgbm.cv_results_['mean_train_score'][k] for k
        param_test.append(-(np.mean([bayes_lgbm.cv_results_['mean_test_score'][k] for k in
    plt.plot(param_set, param_train, label='Train')
    if(type(param_set[0]).__name__ != 'str'):
        plt.plot(param_set, np.poly1d(np.polyfit(param_set, param_train, 1))(param_set), '
    plt.plot(param_set, param_test, label='Test')
    if(type(param_set[0]).__name__ != 'str'):
        plt.plot(param_set, np.poly1d(np.polyfit(param_set, param_test, 1))(param_set), '-
    plt.xlabel(i)
    plt.ylabel('Average RMSE')
    plt.title('Effect of Hyperparameter %s on Bayesian Optimization' %i)
    plt.legend()
    plt.show()
```

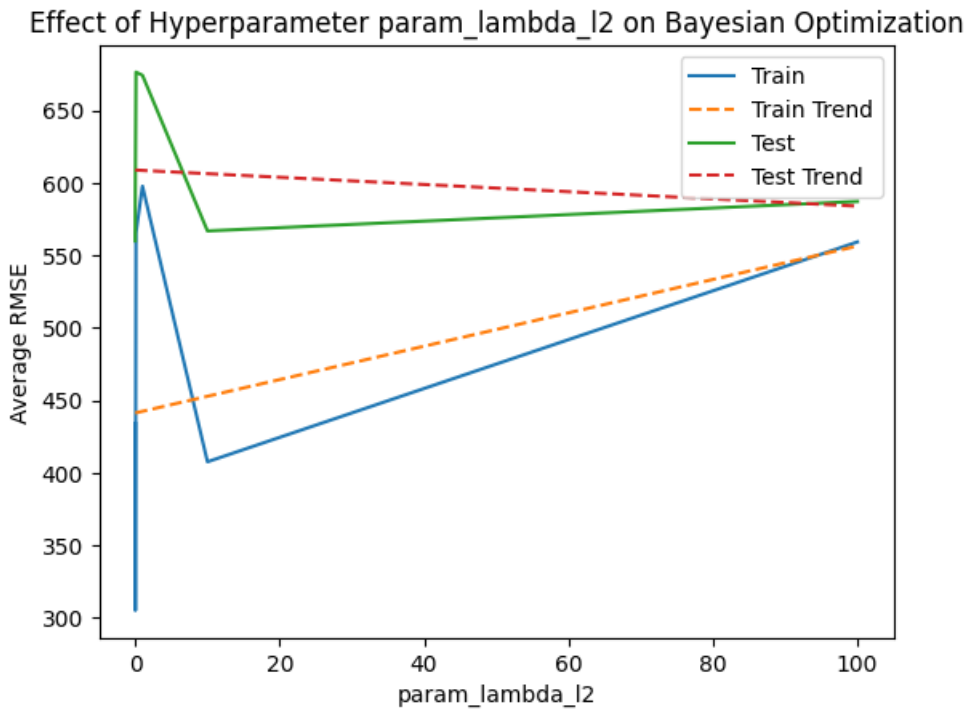


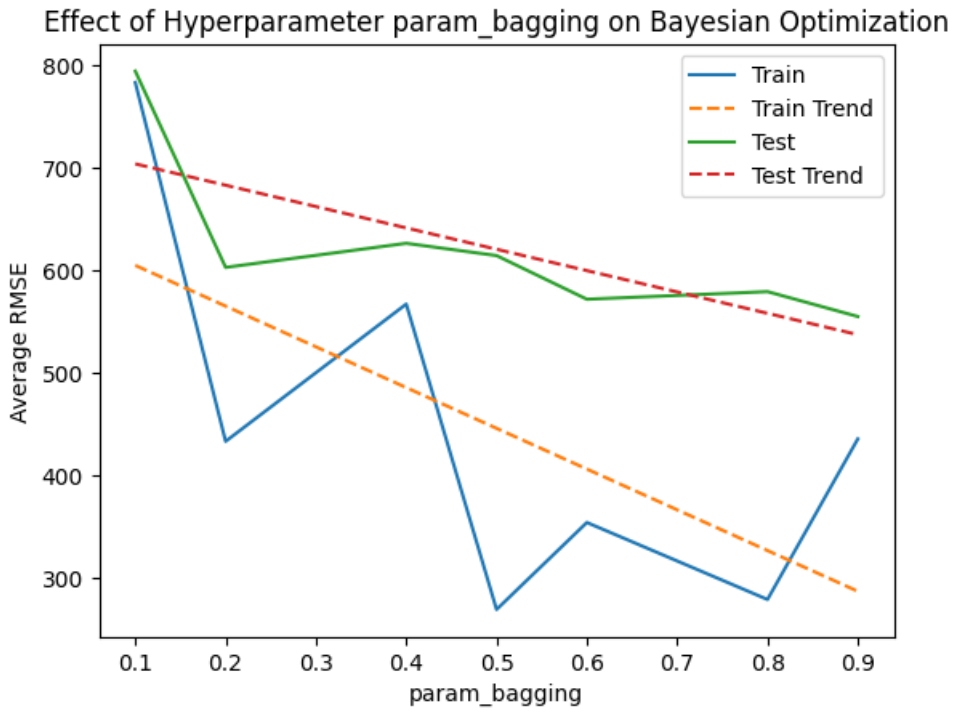


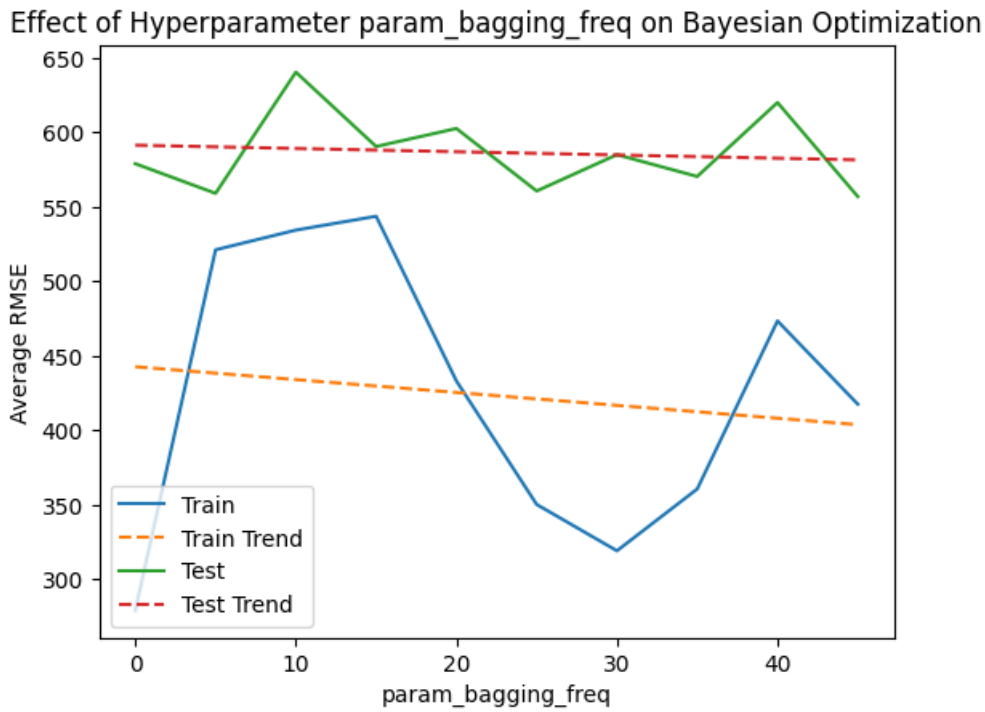

Effect of Hyperparameter param_num_leaves on Bayesian Optimization
Train
Train Trend
Test
Test Trend
Average RMSE
800
700
600
500
400
300
200
0
200
400
600
800
param_num_leaves

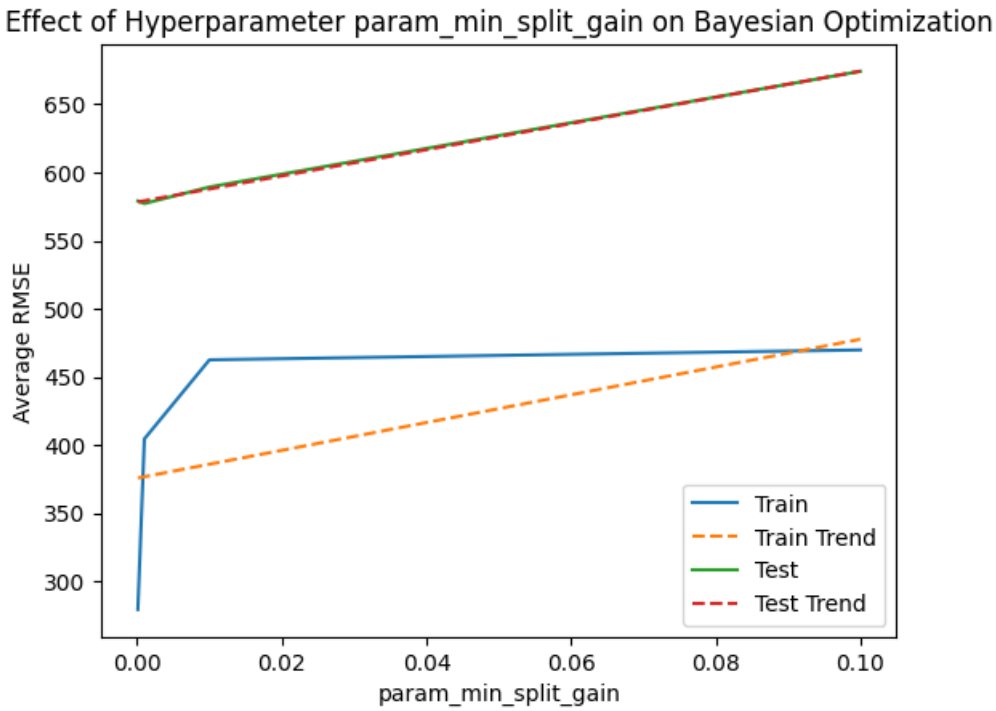

Effect of Hyperparameter param_max_depth on Bayesian Optimization
Train
Train Trend
Test
Test Trend
Average RMSE
700
600
500
400
300
200
5
10
15
20
25
30
35
40
param_max_depth


Effect of Hyperparameter param_lambda_l1 on Bayesian Optimization
Train
Train Trend
Test
Test Trend
Average RMSE
800
700
600
500
400
300
0
20
40
60
80
100
param_lambda_l1


Effect of Hyperparameter param_lambda_l2 on Bayesian Optimization
Train
Train Trend
Test
Test Trend
Average RMSE
650
600
550
500
450
400
350
300
0
20
40
60
80
100
param_lambda_l2


Effect of Hyperparameter param_bagging on Bayesian Optimization
Train
Train Trend
Test
Test Trend
Average RMSE
800
700
600
500
400
300
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
param_bagging


Effect of Hyperparameter param_bagging_freq on Bayesian Optimization
Train
Train Trend
Test
Test Trend
Average RMSE
650
600
550
500
450
400
350
300
0
10
20
30
40
param_bagging_freq


Effect of Hyperparameter param_min_split_gain on Bayesian Optimization
Average RMSE
650
600
550
500
450
400
350
300
0.00
0.02
0.04
0.06
0.08
0.10
param_min_split_gain
Train
Train Trend
Test
Test Trend

P4_Q9

March 9, 2023

[1]:
```
import datetime
import json
import os
import pytz

import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

from tqdm.auto import tqdm
```

0.0.1 Question 9.1

Report the following statistics for each hashtag, i.e. each file has: - Average number of tweets per hour - Average number of followers of users posting the tweets per tweet (to make it simple, we average over the number of tweets; if a users posted twice, we count the user and the user's followers twice as well) - Average number of retweets per tweet

[2]:
```
def read_file(filename: str):
    data_dir = "/tmp2/kch/ece219/ECE219_tweet_data"
    data = []
    n = 100

    with open(os.path.join(data_dir, filename)) as f:
        for line in tqdm(f):
            all_info = json.loads(line)
            taken_info = {
                "type": all_info["type"],
                "title": all_info["title"],
                "text": all_info["tweet"]["text"],
                "id": all_info["tweet"]["id"],
                "hashtags": [
                    h["text"].lower() for h in␣
 ↪all_info["tweet"]["entities"]["hashtags"]
                ],
                "retweet_count": all_info["tweet"]["retweet_count"],
                "author_followers": all_info["author"]["followers"],
```

```
                "original_author_followers":␣
↪all_info["original_author"]["followers"],
                "citation_date": all_info["citation_date"],
                "ranking_score": all_info["metrics"]["ranking_score"],
                "citations": all_info["metrics"]["citations"]["total"],
                "impressions": all_info["metrics"]["impressions"],
            }
            data.append(taken_info)
    #         if len(data) == n:
    #             break

    data_df = pd.DataFrame(data)
    return data_df

def get_filepaths(rootdir: str):
    res = []
    for subdir, dirs, files in os.walk(rootdir):
        for file in files:
            filepath = subdir + os.sep + file
            if filepath.endswith(".txt"):
                res.append(filepath)
    return res

def file_statistics(df: str):
    pst_tz = pytz.timezone('America/Los_Angeles')
    df['citation_date_datetime'] = df['citation_date'].apply(datetime.datetime.
↪fromtimestamp, args=(pst_tz,))
    max_citation_date = max(df['citation_date_datetime'])
    min_citation_date = min(df['citation_date_datetime'])

    avg_tweets_per_hour = len(df) / ((max_citation_date - min_citation_date).
↪total_seconds() / 60 / 60)
    avg_followers_posting_tweets = df['citations'].sum() / len(df)
    avg_retweet_per_tweet = df['author_followers'].sum() / len(df)

    print(f'Average tweets per hour: {avg_tweets_per_hour}')
    print(f'Average number of users posting tweets per tweet:␣
↪{avg_followers_posting_tweets}')
    print(f'Average number of retweet per tweet: {avg_retweet_per_tweet}')
```

[3]:
```
rootdir = "/tmp2/kch/ece219/ECE219_tweet_data"
dataset = {}

filepaths = get_filepaths(rootdir)
dataset_dict = {filepath: read_file(filepath) for filepath in filepaths}
```

```
0it [00:00, ?it/s]
```

```
0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]
```

[4]:
```python
for filepath in filepaths:
    print(filepath)
    file_statistics(dataset_dict[filepath])
    print()
```

```
/tmp2/kch/ece219/ECE219_tweet_data/tweets_#superbowl.txt
Average tweets per hour: 2072.1184017040796
Average number of users posting tweets per tweet: 2.3911895819207736
Average number of retweet per tweet: 8814.96799424623

/tmp2/kch/ece219/ECE219_tweet_data/tweets_#sb49.txt
Average tweets per hour: 1276.8570598680474
Average number of users posting tweets per tweet: 2.52713444111402
Average number of retweet per tweet: 10374.160292019487

/tmp2/kch/ece219/ECE219_tweet_data/tweets_#gohawks.txt
Average tweets per hour: 292.48785062173687
Average number of users posting tweets per tweet: 2.0132093991319877
Average number of retweet per tweet: 2217.9237355281984

/tmp2/kch/ece219/ECE219_tweet_data/tweets_#nfl.txt
Average tweets per hour: 397.0213901819841
Average number of users posting tweets per tweet: 1.5344602655543254
Average number of retweet per tweet: 4662.37544523693

/tmp2/kch/ece219/ECE219_tweet_data/tweets_#gopatriots.txt
Average tweets per hour: 40.95469800606194
Average number of users posting tweets per tweet: 1.4081919101697078
Average number of retweet per tweet: 1427.2526051635405

/tmp2/kch/ece219/ECE219_tweet_data/tweets_#patriots.txt
Average tweets per hour: 750.89426460689
Average number of users posting tweets per tweet: 1.7852871288476946
Average number of retweet per tweet: 3280.4635616550277
```

[ ]:
```python

```

0.0.2 Question 9.2

Plot "number of tweets in hour" over time for #SuperBowl and #NFL (a bar plot with 1-hour bins). The tweets are stored in separate files for different hashtags and files are named as tweet [#hashtag].txt.

[5]:
```
def tweets_per_hour(df: pd.DataFrame):
    return df[['title']].groupby([df['citation_date_datetime'].dt.date,
                                  df['citation_date_datetime'].dt.hour]).count()
```

[6]:
```
def tweets_per_hour_plot(df: pd.DataFrame, title_prefix="", sample_freq=20):

    group_df = tweets_per_hour(df)
    fig = plt.figure(figsize = (10, 5))

    plt.bar([f"{x[0]} {x[1]}H" for x in group_df.index],
            group_df.melt()['value'],
            width=1
           )

    plt.xticks(np.arange(0, len(group_df.index), step=sample_freq),
               [f"{x[0]} {x[1]}H" for x in group_df.index][::sample_freq],
               rotation=90)
    plt.xlabel("Citation Datetime")
    plt.ylabel("Number of Tweets")
    plt.title(f"{title_prefix} Number of Tweets in a Hour over Time")
    plt.show()
```

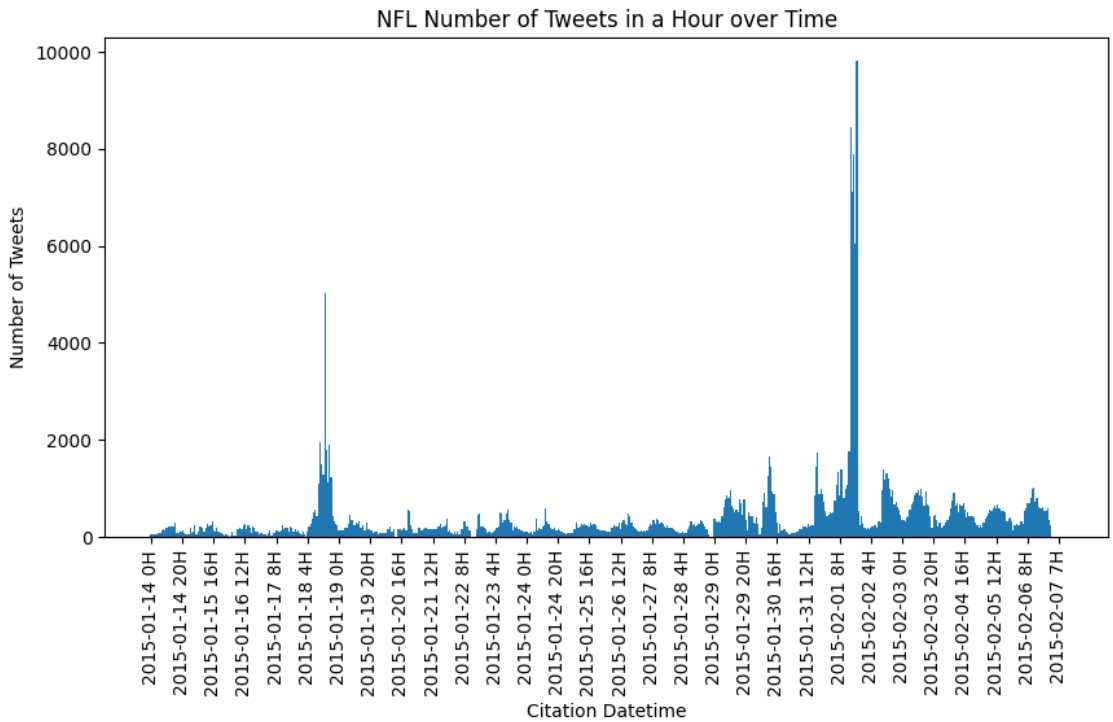
[7]:
```
dataset_dict[f"{rootdir}/tweets_#nfl.txt"]
```

[7]:
```
                  type                                              title  \
0       retweet:native  Who do you have?!?! #nfl #NFLPlayoffs #Packers…
1       retweet:native  http://t.co/H5JADypiEB #billbelichick #NFL #NF…
2       retweet:native  One more week until the #Seahawks begin the #N…
3       retweet:native  We have NFLSHOP on our site! 3% cash back and …
4       retweet:native  Most @SuperBowl wins: #Steelers (6), #49ers &a…
…                  …                                                …
233017           tweet  El Cajas y sus hermosos habitantes  #AllYouNee…
233018           tweet  Watching #nfl network. Replaying #sb49. Never …
233019           tweet  Four players were fined for the brawl late in …
233020           tweet  Huge touchdown by our guy @dannyamendola Don't…
233021           tweet  Big time takeaway by @Seahawks! Make sure you …

                                                     text                  id  \
0       Who do you have?!?! #nfl #NFLPlayoffs #Packers…  550144662250729472
1       http://t.co/H5JADypiEB #billbelichick #NFL #NF…  550872449974558720
2       One more week until the #Seahawks begin the #N…  551827857262280704
3       We have NFLSHOP on our site! 3% cash back and …  552225977745833984
```

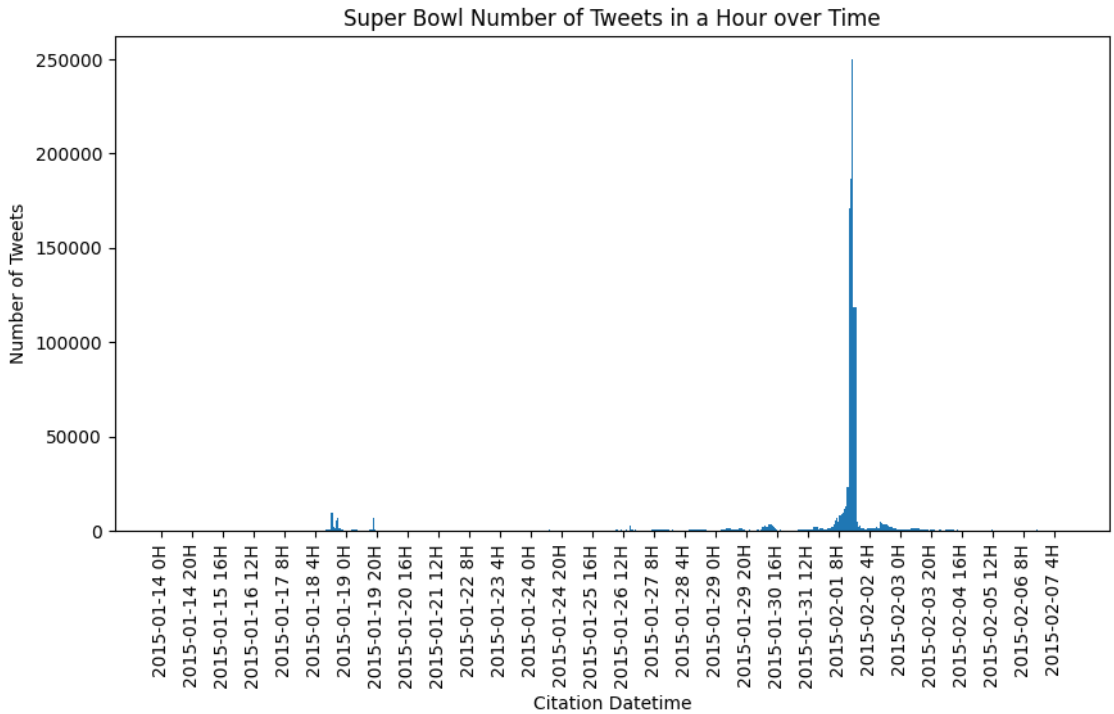
```
4       Most @SuperBowl wins: #Steelers (6), #49ers &a…  552561374577299458
…                                                   …                   …
233017  El Cajas y sus hermosos habitantes  #AllYouNee…  564124624263585792
233018  Watching #nfl network. Replaying #sb49. Never …  564127112316350464
233019  Four players were fined for the brawl late in …  564130766901477377
233020  Huge touchdown by our guy @dannyamendola Don't…  564135259777028096
233021  Big time takeaway by @Seahawks! Make sure you …  564135403708776448

                                                hashtags  retweet_count  \
0       [nfl, nflplayoffs, packers, patriots, broncos,…              0
1       [billbelichick, nfl, nflplayoffs, patsnation, …              0
2                                [seahawks, nfl, hawks]              0
3              [nfl, cowboys, broncos, patriots, packers]              2
4       [steelers, 49ers, cowboys, packers, giants, ra…              0
…                                                   …            …
233017                  [allyouneedisecuador, nfl, sb49]              0
233018                                       [nfl, sb49]              0
233019                                       [nfl, sb49]              0
233020                                [sb49, nflmobile]              0
233021                                [sb49, nflmobile]              0

        author_followers  original_author_followers  citation_date  \
0                   41.0                      176.0     1421517546
1                  361.0                     1611.0     1421258906
2                    6.0                      306.0     1421518663
3                 1364.0                      181.0     1421380685
4                  580.0                    36489.0     1421257471
…                  …                        …            …
233017            1589.0                     1589.0     1423332766
233018             391.0                      391.0     1423333359
233019             360.0                      360.0     1423334230
233020              69.0                       69.0     1423335301
233021              69.0                       69.0     1423335336

        ranking_score  citations  impressions    citation_date_datetime
0            6.352194          4           42 2015-01-17 09:59:06-08:00
1            4.211227          2          272 2015-01-14 10:08:26-08:00
2            3.417832          2            4 2015-01-17 10:17:43-08:00
3            6.418528          2         1365 2015-01-15 19:58:05-08:00
4            3.915142         14          564 2015-01-14 09:44:31-08:00
…               …        …           …                        …
233017       3.819176          1         1563 2015-02-07 10:12:46-08:00
233018       4.184327          1          386 2015-02-07 10:22:39-08:00
233019       3.945940          1          200 2015-02-07 10:37:10-08:00
233020       3.823271          1           34 2015-02-07 10:55:01-08:00
233021       3.863007          1           34 2015-02-07 10:55:36-08:00
```

```
[233022 rows x 13 columns]
```

[8]:
```
tweets_per_hour_plot(dataset_dict[f"{rootdir}/tweets_#nfl.txt"],␣
↪title_prefix="NFL")
```



[9]:
```
tweets_per_hour_plot(dataset_dict[f"{rootdir}/tweets_#superbowl.txt"],␣
↪title_prefix="Super Bowl")
```


Super Bowl Number of Tweets in a Hour over Time
Number of Tweets
250000
200000
150000
100000
50000
0
2015-01-14 0H
2015-01-14 20H
2015-01-15 16H
2015-01-16 12H
2015-01-17 8H
2015-01-18 4H
2015-01-19 0H
2015-01-19 20H
2015-01-20 16H
2015-01-21 12H
2015-01-22 8H
2015-01-23 4H
2015-01-24 0H
2015-01-24 20H
2015-01-25 16H
2015-01-26 12H
2015-01-27 8H
2015-01-28 4H
2015-01-29 0H
2015-01-29 20H
2015-01-30 16H
2015-01-31 12H
2015-02-01 8H
2015-02-02 4H
2015-02-03 0H
2015-02-03 20H
2015-02-04 16H
2015-02-05 12H
2015-02-06 8H
2015-02-07 4H
Citation Datetime

[ ]:

```
```

SuperBowl Leading Status Classification using Fine-tuned Bert Transformer with Numerous Feature Extraction

Kuei-Chun Huang
705668757
kueichun0625@g.ucla.edu

Sinya Peng
805861373
sinya@g.ucla.edu

Yu-Wei Chen
205928946
yuweichen@g.ucla.edu

1 Task Definition

In this project, we tackled the time-series correlation between scores and tweets problem. We notice the tweet is real-time, reflecting the game status. When the score changes, the tweet can act as the signal, which implies which team stays ahead of the pack. We formula this question as a three-class classification problem including "Patriots is leading", "Seahawks is leading", and "tie". Our goal is to predict which team takes the lead at that moment when a given tweet is generated. We extract numerous features such as emojis, hashtag text, and user location to support the prediction task. We also tried different embedding methods like the term frequency-inverse document frequency (TF-IDF) and bidirectional encoder representations from transformers (Bert). We conduct our experiments on the SuperBowl XLIX tweet dataset. Our proposed approach achieved accuracy over 0.63 when predicting the leading status. Further, we use a generative model to generate the tweet using an ensemble of sentences from the original data. The experimental results show that the generative model is able to generate a human-readable tweet based on the tweets while the score change.

2 Data Exploration

Table 1: SuperBowl XLIX scoring summary.

#	Quarter	Time Left in Quarter	Time	Patriots Score	Seahawks Score
1	2ND	9:47	16:12:41	7	0
2	2ND	2:16	16:33:53	7	7
3	2ND	0:31	16:47:58	14	7
4	2ND	0:02	16:58:26	14	14
5	3RD	11:09	17:37:51	14	17
6	3RD	4:54	17:53:52	14	24
7	4TH	7:55	18:27:18	21	24
8	4TH	2:02	18:49:20	28	24

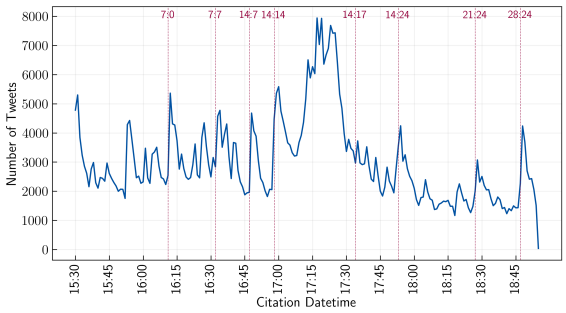


Figure 1: Number of #superbowl tweets in a minute with the scoring time point (red).

First, we acquire all the scoring moments from the historical data. By doing so, we can obtain the ground truth of the leading status. There are eight scoring time points, and their absolute time is shown in table 1. We map the scoring time points and the number of tweets in figure 1. Clearly, we saw the peak of the number of tweets can correspond to each scoring time point. The only peak not caught by the scoring time point is the halftime show held around 17:20. Further, an interesting finding is that when the Patriots score, we can observe a peak in the number of #gopatriots hashtag tweets as shown in figure 2.

However, the number of the #gohawks hashtag is not reflecting the same trend. This might be because the hashtag #gohawks are posted by the Seahawks fans. The Seahawks fans might not be pleased to see the enemy team gain score. The same observation can be found in the #gohawks hashtag histogram as well in figure 3. We can see that when different teams score goals, there is an interaction that ebbs and flows.

Further, when we look into the potentially useful information in tweets. The location of the tweet

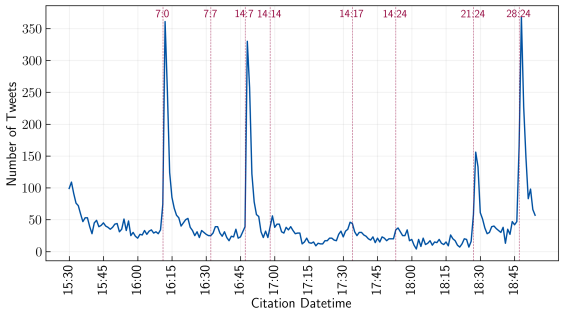


Figure 2: Number of #gopatriots hashtag tweets in a minute with the scoring time point (red).

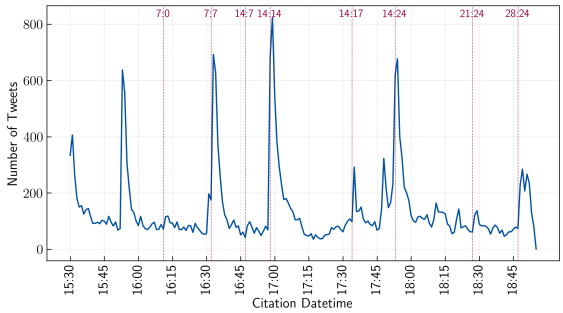


Figure 3: Number of #gohawks hashtag tweets in a minute with the scoring time point (red).

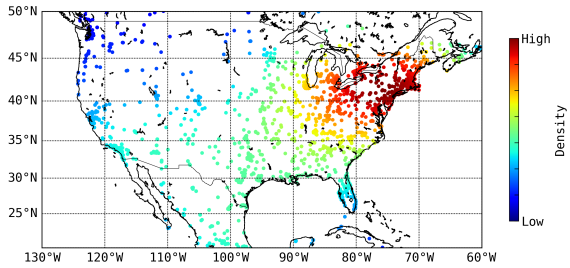


Figure 4: #gopatriots hashtag tweets scatter plot colored by density.

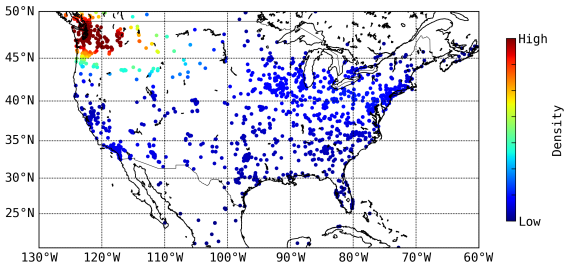


Figure 5: #gohawks hashtag tweets scatter plot colored by density.

seems to be useful. However, extracting and plotting the tweet's location is quite challenging since the format of the location saved in the tweet data is not standard and messy. First, the way users input their locations can vary, even when referring to the same place, and sometimes the input may not even represent a real location. Therefore, it is necessary to verify the accuracy of the location by using a geocoding service. To accomplish this, the GeoPy library, which acts as a client for several commonly used geocoding web services, will be utilized. Consequently, for each of the user-provided locations, GeoPy will be invoked to obtain a standardized and validated location. This will ensure that all locations are presented in a consistent format, and any invalid inputs are identified.

Secondly, we use the gaussian kde (Parzen, 1962; Hastie et al., 2009) to estimate the density of the cluster to plot the heatmap. A kernel density estimation is a non-parametric method used to approximate the probability density function (PDF) of a random variable. The gaussian kde algorithm is suitable for univariate and multivariate datasets and incorporates automated bandwidth determination.

Finally, we use the Basemap to plot map projections with coastlines and political boundaries using Matplotlib. The #gopatriots hashtag tweets scatter plot colored by density can be found in 4. It can be clearly seen that the #gopatriots tweets scatter on the east coast, whereas the #gohawks tweets scatter in Seattle. This also reflects the practical geographical relationship of these teams. This also indicates even if the tweet did not include the "#gopatriots" or "#gohawks" tag. We can still identify their preference based on their location and help with the classification task.

3 Feature Engineering

From the data preprocessing perspective, we implemented symbol replacement and entities-related regex to make the tweet sentence more general. Further, we implemented the hashtag split to add the hashtag text as extra information to help the task. Next, we implemented the sentiment analysis (Bravo-Marquez et al., 2013; Bollen et al., 2011; Herzallah et al., 2018) and used the sentiment as a feature. Moreover, we demojize the emoji to extract the physical meaning of the emoji symbols.

3.1 Special Character

There are four strategies we designed for data cleaning and preprocessing. The basic strategy is to replace punctuation with space. For example, "Hello, World!" to "Hello World" to unify the appeared vocabulary. The advanced version of replacement is to replace those patterns that usually appear in

Table 2: Examples of the polarity of tweets.

Tweet Text	Sentiment Label	Defined Sentiment Score
7 0 Yay :D #SuperBowlSuper	Extreme Positive	3
Touchdown! #GoPats #NEvsSEA #SuperBowlXLIX	Positive	1
So when is Katy Perry gonna be on TV? #SuperBowlXLIX	Neutral	0
Simon says you shouldn't even be in this game. Smh! #SuperBowlXLIX	Negative	-1
Cḿon Seattle Defense WTF are y'all doing? #SuperBowlXLIX	Extreme Negative	-3

sentences. For example, while 145$ appears in a sentence, we can replace it with $#money#$. The dollar sign $ and hash sign # added is to differentiate from normal money vocabulary. Besides money, we design several patterns like Date, Time, and Text with Numbers. Also, for punctuation, we suppose some of the punctuation has meaning in a sentence. For example, the exclamation mark! It can represent the meaning of surprise. Also, we observe that the number of exclamation marks relates to the degree of emotion.

3.2 Hashtag Text

The further step is that split the hashtag in the Twitter text. The intention is that it is common for Twitter to have hashtags like #BailoutHumans and #BailoutHumansNow. These hashtags are supposed to be similar; however, it is not detected using a word-level tokenizer (the char-level is capable of dealing with this case). Therefore, we add a preprocessing rule which splits this phrase into common vocabularies. In this way, #BailoutHumans will have more similar embedding with #BailoutHumansNow compare to the previous method.

3.3 Sentiment

In this step, we use sentiment information. The intention comes from the statistics see table 2. Table 2 shows the emotion categories which have the five different sentiment scores with the corresponding tweet. The sentiment scores are generated from the sentiment analysis model, which classifies the new text's polarity into positive and negative probabilities. Ideally, we expect the text classification model can learn the sentiment associatively; however, different text models have different expertise. This is not a feature that comes with all text models. We classify the text into extreme_positive, positive, neutral, negative, and extreme_negative, and add it as a feature using the self-defined corresponding score.

3.4 Demojize

In this step, we extract information from emojis. For example, if we have emojis like ":)" it means a smiling face. We then turn those emojis into text. In this way, the physical meaning of the emoji symbols can also be considered in the following training text model.

4 Methods

For the embedding generation, we implement the term TF-IDF, pre-trained Sentence Transformer, and fine-tuned BERT. For the dimension reduction, we try NMF and LSI with a fixed number of components 80. For the classification method, we choose SVM with default hyper-parameters for TF-IDF and pre-trained Sentence Transformer embeddings, while for fine-tuned BERT, we use the pre-defined Transformer for Sequence Classification model architecture.

4.1 MPNET Base Model

MPNET base model is known to have a solid sentence-level understanding. The sentence transformers model is capable of transforming sentences and paragraphs into a 768 dimensional dense vector space, making it useful for tasks such as semantic search or clustering. Therefore, we replace dimension reduced TF-IDF embeddings with embeddings generated by a pre-trained Sentence Transformer and feed them into the downstream SVM classifier.

4.2 BERT Base Model with Fine-Tuning

BERT is a pre-trained language model that uses a bidirectional transformer encoder to generate contextualized word embeddings. The base model is the smaller and more computationally efficient version of the BERT model. Fine-tuning is a technique used to adapt pre-trained models to specific downstream tasks by training them on task-specific datasets.

BERT base model with fine-tuning involves taking the pre-trained BERT base model and fine-tuning it on a downstream task. The process in-

Table 3: Comparison of the model performances.

Model		Preprocess	Additional Feature	Accuracy	F1 Score		
					Patriots	Seahawks	Tie
Baselines	Zero Rate			0.48122	0.	0.	0.64975
	Rule Based			0.49396	0.21118	0.19597	0.65457
	TF-IDF + LSI + SVM			0.55130	0.32677	0.38152	0.68587
TF-IDF Based	TF-IDF + LSI + SVM	Stemming		0.54750	0.32192	0.37629	0.68369
	TF-IDF + LSI + SVM	Normalization		0.55463	0.33492	0.39058	0.68763
	TF-IDF + LSI + SVM		Location	0.54355	0.29843	0.37047	0.68239
	TF-IDF + LSI + SVM		Sentiment	0.55152	0.32678	0.38243	0.68599
	TF-IDF + NMF + SVM			0.54817	0.31408	0.36869	0.68337
	TF-IDF + LSI + SVM	Normalization	Sentiment	**0.55468**	0.33507	0.39078	0.68760
Transformer Based	MPNET Base Model + LSI + SVM	Normalization	Sentiment	0.60507	0.44236	0.50269	0.71996
	BERT Base Model with Fine-Tuning	Normalization		**0.63017**	0.46979	0.55564	0.73832
Human Predictor				0.59000	0.56250	0.51429	0.63366

volves adding a task-specific layer on top of the pre-trained BERT model and training it on a labeled dataset for the specific task. During training, the weights of the pre-trained BERT model are adjusted along with the task-specific layer, allowing the model to learn task-specific features and improve its performance on the downstream task. Once the fine-tuning is complete, the resulting model can be used to make predictions on new, unseen data for the specific task.

Therefore, to make better use of the power of Transformers, we further fine-tuned the weights from a pre-trained BERT on our task. Since it is a classification task, we directly utilize `BERTForSequenceClassification` model.

4.3 Textgenrnn Model

Textgenrnn is a deep learning-based text generation model that uses a Recurrent Neural Network (RNN) architecture to generate text. It is capable of generating coherent and meaningful sentences, paragraphs, and even longer pieces of text based on a given input text.

The model works by training on a dataset of text and learning to predict the next word in a sequence of words, given the preceding words. The RNN architecture of the model allows it to take into account the context and generate text that is coherent and consistent with the input text. The training process involves updating the weights of the model based on the error between the predicted word and the actual next word in the sequence.

Therefore, we trained the Textgenrnn model by feeding the tweet while the score change to make the model generate the corresponding tweet.

5 Evaluation

Finally, we present the experiment in tweet #superbowl data. We also discuss on how the generative model generates a tweet using an ensemble of sentences from the original data and is also performed at last. Through empirical evaluation of tweet #superbowl data, the tweet text is shown to provide effective and efficient performance in predicting the leading status given the tweet. In particular, the experiment results indicate that the fine-tuned transformer model outperforms all other methods including human predictor and can reach 63% accuracy.

5.1 Baselines

We chose three related models as baselines to compare with our model. One of them is based on TF-IDF acting as the strong baseline, and the rest two are traditional methods acting as the weak baselines.

- Zero Rate: The Zero Rate Classifier is a simple classification algorithm that predicts the most frequent class for all instances in the dataset. Although the Zero Rate Classifier is not sophisticated, it can serve as a baseline for evaluating other models' performance. In imbalanced datasets, where one class is much more prevalent than the others, a model that performs slightly better than Zero Rate Classifier can still be valuable. However, for a model to be considered useful, it must outperform the Zero Rate Classifier.

- Rule Based: The Rule Based method classifies the tweet based on if the tweet mentioned the team keywords. For the keyword pool, we include the nicknames, players, and regions of the specific team. For example, if

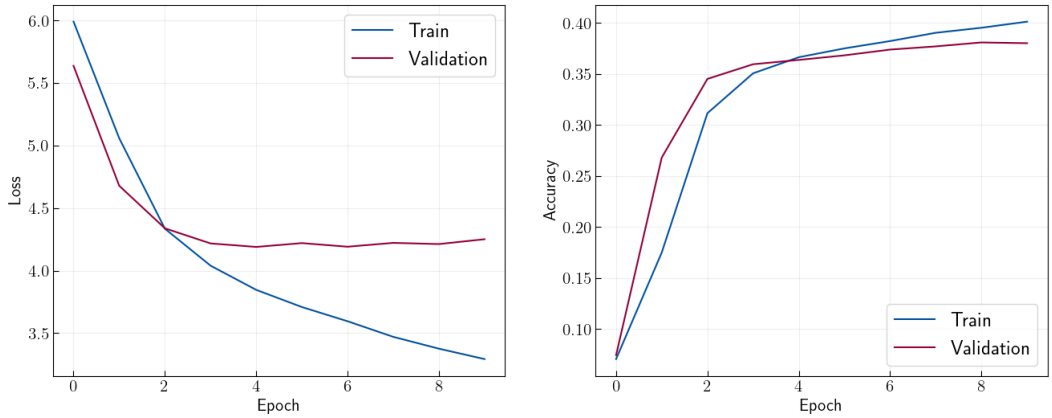


Figure 6: Textgenrnn model loss and accuracy.

Table 4: Examples of the polarity of tweets.

Score Change	Generated Tweet Text
Patriots	someone hurt i need to use the pats so #gopatriots
	nice to finish go with this ? ! #ontoseattle #pepsisbhalftime
Seahawks	1 - 0 Seattle over Boston ; be safe and got this backs #gohawks
	from up at home today, it's been crazy was there. #gohawks

"Tom Brady" is mentioned in the tweet. The tweet is more likely to talk about the game status/scoring of the Patriots, and the Rule-Based method would output Patriots as the prediction result.

- Naïve TF-IDF + LSI + SVM: The input of this pipeline is features without any processing. We set it as our strong baseline.

5.2 Human Predictor

We wanted to know how good a human can do on this task. So we randomly drew 100 instances from the test set for our team members to annotate and evaluate the result with the same metrics. In particular, this predictor was supposed to be our upper bound.

5.3 Results

The evaluation results are shown in table 3. As we can see in the table, our two weak baselines have accuracies of 0.48122 and 0.49396, while our strong baseline can reach 0.55130. Furthermore, our best TF-IDF-based model, which includes text normalization, sentence sentiment feature, and LSI as dimension reducer, slightly outperforms the strong baseline with an accuracy of 0.55468. Finally, both our two Transformer-based models significantly beat the strong baseline and the human predictor with accuracies of 0.60507 and 0.63017.

5.4 Generative Model

From the experiment, we notice that Textgenrnn is designed to generate text that is similar to what we might see in real tweets. Still, the generated tweets are not always making complete sense or be entirely grammatically correct. Additionally, the content of the tweets is based solely on patterns learned from the training data and does not reflect the opinions or thoughts of any particular person or group; therefore, the tweets generated by Textgenrnn may resemble real tweets.

The training and validation accuracy and loss of the Textgenrnn model are shown in figure 6. Table 4 show examples of tweet generated by Textgenrnn for the New England Patriots and the Seattle Seahawks when the score changes. Although those generated tweets are not fully human-readable, the main preference of each tweet is clear and obvious.

6 Discussion

First of all, we designed a rule-based model, which we expected to have a much better performance than the Zero-rate model. However, it turned out to only improve a bit. Based on this result, we found that this task required an even more complicated model to capture the relationship between a tweet and the leading status. Therefore, we built a pipeline of what we have learned from class as our

strong baseline and tried to improve it.

Given past experience from previous homework, applying stemming to the input text and using NMF as a dimension reducer tend to lead to better performance. But unfortunately, as we can see in table 3, both of them harm the model performance a lot. In addition, we thought that the location would help the classification based on our observation from data exploration, but unexpectedly, adding the location feature will also harm the classification accuracy. We guessed that it is because whether leading or trailing, the fans of both teams are almost enthusiastic, which makes the location feature even noisier.

Frustrating as the result was, we still tried to improve our TF-IDF-based model and finally found that applying several normalizations on the input text and adding a sentiment feature can slightly improve the performance. After lots of work, we assumed that it was the best a TF-IDF-based model could do. We thus attempted to replace the text encoder with a Transformer. It is interesting to note that the performance was dramatically boosted just by substituting the TF-IDF embeddings with ones generated by a pre-trained Transformer, which shows the capability to capture essential information from a tweet so that the model even outperforms the human predictor.

With such an exciting result, we further fine-tuned a pre-trained Transformer on our dataset. As expected, with a powerful language model as the tweet feature extractor, there is another significant performance jump.

References

Johan Bollen, Huina Mao, and Alberto Pepe. 2011. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *Fifth International AAAI Conference on Weblogs and Social Media*.

Felipe Bravo-Marquez, Marcelo Mendoza, and Barbara Poblete. 2013. Combining strengths, emotions and polarities for boosting twitter sentiment analysis. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, pages 1–9.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. 2009. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

Wafa Herzallah, Hossam Faris, and Omar Adwan. 2018. Feature engineering for detecting spammers on twitter: Modelling and analysis. *Journal of Information Science*, 44(2):230–247.

Emanuel Parzen. 1962. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076.

P4_Q10_EDA_Hawks

March 9, 2023

[1]:
```
import datetime
import json
import math
import os
import pytz
import random
import re

from functools import reduce, partial
from joblib import Memory
from tempfile import mkdtemp
from typing import List

import nltk
import numpy as np
import pandas as pd
import umap.umap_ as umap
import umap.plot

import emoji
from matplotlib import pyplot as plt
from nltk import pos_tag
from nltk.corpus import wordnet
from nltk.stem import PorterStemmer, WordNetLemmatizer
from nltk.tokenize import word_tokenize
from sklearn import preprocessing, metrics
from sklearn.base import TransformerMixin, BaseEstimator
from sklearn.compose import ColumnTransformer
from sklearn.decomposition import NMF, TruncatedSVD
from sklearn.feature_extraction.text import CountVectorizer, TfidfTransformer
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import Pipeline
from sklearn.metrics import (accuracy_score,
                             auc,
                             confusion_matrix,
                             ConfusionMatrixDisplay,
                             f1_score,
```

```
                              precision_score,
                              RocCurveDisplay,
                              recall_score,
                              roc_curve,
                             )
from sklearn.model_selection import GridSearchCV, train_test_split
from sklearn.multiclass import OneVsRestClassifier, OneVsOneClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import LinearSVC
from sklearn.utils.extmath import randomized_svd
from textblob import TextBlob
from tqdm.auto import tqdm
```

```
2023-03-08 05:24:10.162455: I tensorflow/core/platform/cpu_feature_guard.cc:193]
This TensorFlow binary is optimized with oneAPI Deep Neural Network Library
(oneDNN) to use the following CPU instructions in performance-critical
operations:  AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate
compiler flags.
2023-03-08 05:24:10.287148: W
tensorflow/compiler/xla/stream_executor/platform/default/dso_loader.cc:64] Could
not load dynamic library 'libcudart.so.11.0'; dlerror: libcudart.so.11.0: cannot
open shared object file: No such file or directory; LD_LIBRARY_PATH:
/usr/local/cuda-9.0/lib64:/usr/local/cuda-9.0/lib64:
2023-03-08 05:24:10.287187: I
tensorflow/compiler/xla/stream_executor/cuda/cudart_stub.cc:29] Ignore above
cudart dlerror if you do not have a GPU set up on your machine.
2023-03-08 05:24:11.227962: W
tensorflow/compiler/xla/stream_executor/platform/default/dso_loader.cc:64] Could
not load dynamic library 'libnvinfer.so.7'; dlerror: libnvinfer.so.7: cannot
open shared object file: No such file or directory; LD_LIBRARY_PATH:
/usr/local/cuda-9.0/lib64:/usr/local/cuda-9.0/lib64:
2023-03-08 05:24:11.228077: W
tensorflow/compiler/xla/stream_executor/platform/default/dso_loader.cc:64] Could
not load dynamic library 'libnvinfer_plugin.so.7'; dlerror:
libnvinfer_plugin.so.7: cannot open shared object file: No such file or
directory; LD_LIBRARY_PATH: /usr/local/cuda-9.0/lib64:/usr/local/cuda-9.0/lib64:
2023-03-08 05:24:11.228088: W
tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Cannot
dlopen some TensorRT libraries. If you would like to use Nvidia GPU with
TensorRT, please make sure the missing libraries mentioned above are installed
properly.
```

[2]:
```
SEED = 806
np.random.seed(SEED)
# plt.style.use("seaborn-bright")
random.seed(SEED)
```

```
/tmp/ipykernel_27643/3554118125.py:3: MatplotlibDeprecationWarning: The seaborn
styles shipped by Matplotlib are deprecated since 3.6, as they no longer
correspond to the styles shipped by seaborn. However, they will remain available
as 'seaborn-v0_8-<style>'. Alternatively, directly use the seaborn API instead.
  plt.style.use("seaborn-bright")
```

[30]:
```
SEED = 806
np.random.seed(SEED)
# plt.style.use("seaborn-bright")
random.seed(SEED)

import matplotlib as mpl
mpl.rcParams.update(mpl.rcParamsDefault)


import matplotlib.pylab as pylab
params = {'legend.fontsize': 'x-large',
          'figure.figsize': (15, 5),
         'axes.labelsize': 'x-large',
         'axes.titlesize':'x-large',
         'xtick.labelsize':'x-large',
         'ytick.labelsize':'x-large'}
pylab.rcParams.update(params)

fsize = 15
tsize = 18
tdir = 'in'
major = 5.0
minor = 3.0
lwidth = 0.8
lhandle = 2.0
plt.style.use('default')
# plt.rcParams['text.usetex'] = True
plt.rcParams.update({
#     "text.usetex": True,
    "font.family": "monospace"
})
plt.rcParams['font.size'] = fsize
plt.rcParams['legend.fontsize'] = tsize
plt.rcParams['xtick.direction'] = tdir
plt.rcParams['ytick.direction'] = tdir
plt.rcParams['xtick.major.size'] = major
plt.rcParams['xtick.minor.size'] = minor
plt.rcParams['ytick.major.size'] = 5.0
plt.rcParams['ytick.minor.size'] = 3.0
plt.rcParams['axes.linewidth'] = lwidth
plt.rcParams['legend.handlelength'] = lhandle
```

```
colors = ['#0051a2', '#97964a', '#ffd44f', '#f4777f', '#93003a']


plt.rcParams[u'figure.figsize'] = [12, 8]
plt.rcParams[u'figure.dpi'] = 300
```

[3]:
```
game_start = 1422833400
game_end = 1422845700
```

[4]:
```
import pytz
pst_tz = pytz.timezone('America/Los_Angeles')
datetime_object_in_pst_timezone = datetime.datetime.fromtimestamp(game_start,␣
 ↪pst_tz)
datetime_object_in_pst_timezone
```

[4]:
```
datetime.datetime(2015, 2, 1, 15, 30, tzinfo=<DstTzInfo 'America/Los_Angeles'
PST-1 day, 16:00:00 STD>)
```

[5]:
```
import pytz
pst_tz = pytz.timezone('America/Los_Angeles')
datetime_object_in_pst_timezone = datetime.datetime.fromtimestamp(game_end,␣
 ↪pst_tz)
datetime_object_in_pst_timezone
```

[5]:
```
datetime.datetime(2015, 2, 1, 18, 55, tzinfo=<DstTzInfo 'America/Los_Angeles'
PST-1 day, 16:00:00 STD>)
```

[6]:
```
data_dir = "/tmp2/kch/ece219/ECE219_tweet_data"
filename = "tweets_#superbowl.txt"
data = []
n = 100

with open(os.path.join(data_dir, filename)) as f:
    for line in tqdm(f):
        all_info = json.loads(line)
        # print(all_info.keys())
        # print(all_info["tweet"]["user"])
        taken_info = {
            "type": all_info["type"],
            # "title": all_info["title"],
            "text": all_info["tweet"]["text"],
            "id": all_info["tweet"]["id"],
            "hashtags": " ".join([
                " ".join([s for s in re.split('([A-Z][a-z]+)', h["text"]) if s])
                for h in all_info["tweet"]["entities"]["hashtags"]
            ]),  # hashtag split
```

```
            "location": all_info["tweet"]["user"]['location'],
            "citation_date": all_info["citation_date"],
        }
        if taken_info["citation_date"] > game_end:
            continue
        if taken_info["citation_date"] <= game_start or taken_info["text"] ==␣
↪"":
            continue
        data.append(taken_info)

data_df = pd.DataFrame(data)
```

```
0it [00:00, ?it/s]
```

[ ]:

```

```

[7]:

```
data_df['citation_date_datetime'] = data_df['citation_date'].apply(datetime.
↪datetime.fromtimestamp, args=(pst_tz,))
```

[8]:

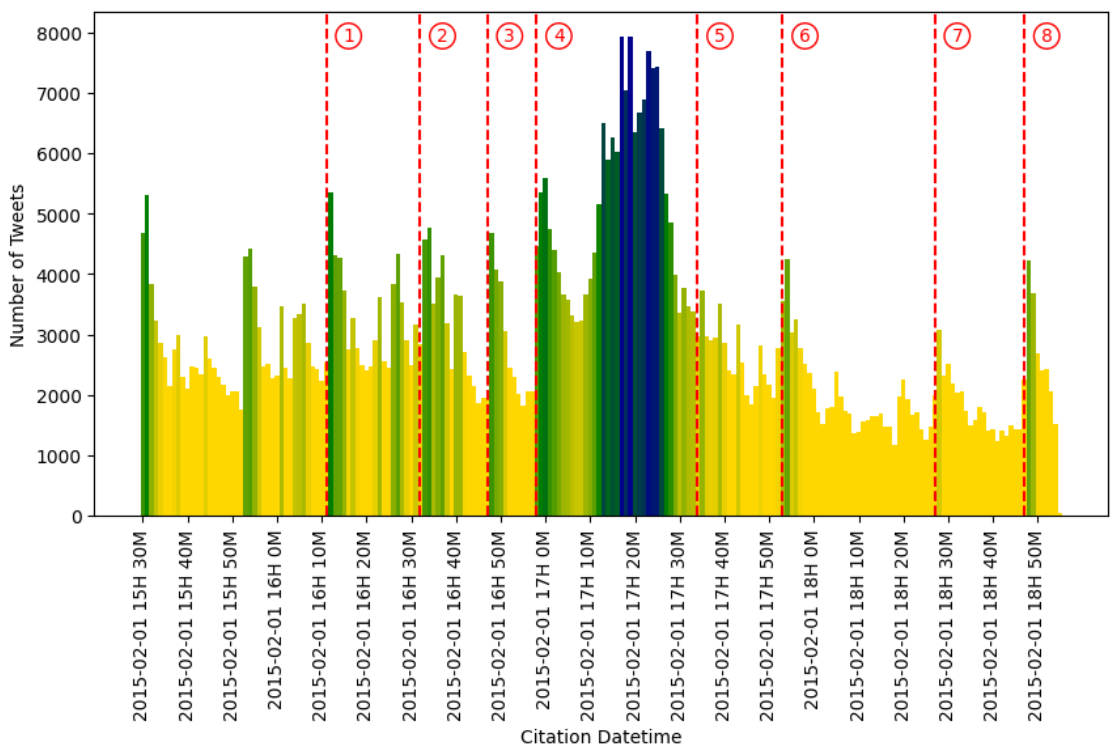
```
def tweets_per_minute(df: pd.DataFrame):
    return df[['text']].groupby([df['citation_date_datetime'].dt.date,
                                 df['citation_date_datetime'].dt.hour,
                                 df['citation_date_datetime'].dt.minute,
                                ]).count()
```

[9]:

```
import matplotlib.colors as mcolors

def tweets_per_minute_plot(df: pd.DataFrame, title_prefix="", sample_freq=10):

    group_df = tweets_per_minute(df)
    fig = plt.figure(figsize = (10, 5))

    cmap = mcolors.LinearSegmentedColormap.from_list("", ["gold", "gold",␣
↪"green", "darkblue"])

    plt.bar([f"{x[0]} {x[1]}H {x[2]}M" for x in group_df.index],
            group_df.melt()['value'],
            width=1,
            color=cmap(group_df.melt()['value']/group_df.melt()['value'].max())
           )
    max_value = (max(group_df.melt()['value']))

    scoring_timestamp = [1422835905, 1422837178, 1422838078, 1422838706,␣
↪1422840875, 1422842032, 1422844038, 1422845270]

    for idx, timestamp in enumerate(scoring_timestamp):
        dt = datetime.datetime.fromtimestamp(timestamp, pst_tz)
```

```
        plt.axvline(list(group_df.index).index((dt.date(), dt.hour, dt.
↪minute)), color="red", linestyle='--')
        plt.text(list(group_df.index).index((dt.date(), dt.hour, dt.minute))+5,
                 max_value+1, str(idx+1),
                 fontsize=10,
                 ha="center",
                 va="center",
                 bbox = dict(boxstyle=f"circle,pad={0.2}", fc="white",␣
↪color="red"), color="red")

    plt.xticks(np.arange(0, len(group_df.index), step=sample_freq),
               [f"{x[0]} {x[1]}H {x[2]}M" for x in group_df.index][::
↪sample_freq],
               rotation=90)
    plt.xlabel("Citation Datetime")
    plt.ylabel("Number of Tweets")
    #plt.title(f"Number of #{title_prefix} tweets in a minute with the scoring␣
↪time point (red)")
    plt.show()
```

[10]:
```
tweets_per_minute_plot(data_df, title_prefix="Super Bowl")
```



[11]:
```
def read_file(filename: str):
    data_dir = "/tmp2/kch/ece219/ECE219_tweet_data"
    data = []
    n = 100

    with open(os.path.join(data_dir, filename)) as f:
        for line in tqdm(f):
            all_info = json.loads(line)
            taken_info = {
                "type": all_info["type"],
                "title": all_info["title"],
                "text": all_info["tweet"]["text"],
                "id": all_info["tweet"]["id"],
                "hashtags": [
                    h["text"].lower() for h in␣
 ↪all_info["tweet"]["entities"]["hashtags"]
                ],
                "retweet_count": all_info["tweet"]["retweet_count"],
                "author_followers": all_info["author"]["followers"],
                "original_author_followers":␣
 ↪all_info["original_author"]["followers"],
                "citation_date": all_info["citation_date"],
                "ranking_score": all_info["metrics"]["ranking_score"],
                "citations": all_info["metrics"]["citations"]["total"],
                "impressions": all_info["metrics"]["impressions"],
            }
            data.append(taken_info)
    #         if len(data) == n:
    #             break

    data_df = pd.DataFrame(data)
    data_df['citation_date_datetime'] = data_df['citation_date'].apply(datetime.
 ↪datetime.fromtimestamp, args=(pst_tz,))
    return data_df

def get_filepaths(rootdir: str):
    res = []
    for subdir, dirs, files in os.walk(rootdir):
        for file in files:
            filepath = subdir + os.sep + file
            if filepath.endswith(".txt"):
                res.append(filepath)
    return res

def file_statistics(df: str):
    pst_tz = pytz.timezone('America/Los_Angeles')
```

```
    df['citation_date_datetime'] = df['citation_date'].apply(datetime.datetime.
↪fromtimestamp, args=(pst_tz,))
    max_citation_date = max(df['citation_date_datetime'])
    min_citation_date = min(df['citation_date_datetime'])

    avg_tweets_per_hour = len(df) / ((max_citation_date - min_citation_date).
↪total_seconds() / 60 / 60)
    avg_followers_posting_tweets = df['citations'].sum() / len(df)
    avg_retweet_per_tweet = df['author_followers'].sum() / len(df)

    print(f'Average tweets per minute: {avg_tweets_per_hour}')
    print(f'Average number of users posting tweets per tweet:␣
↪{avg_followers_posting_tweets}')
    print(f'Average number of retweet per tweet: {avg_retweet_per_tweet}')
```

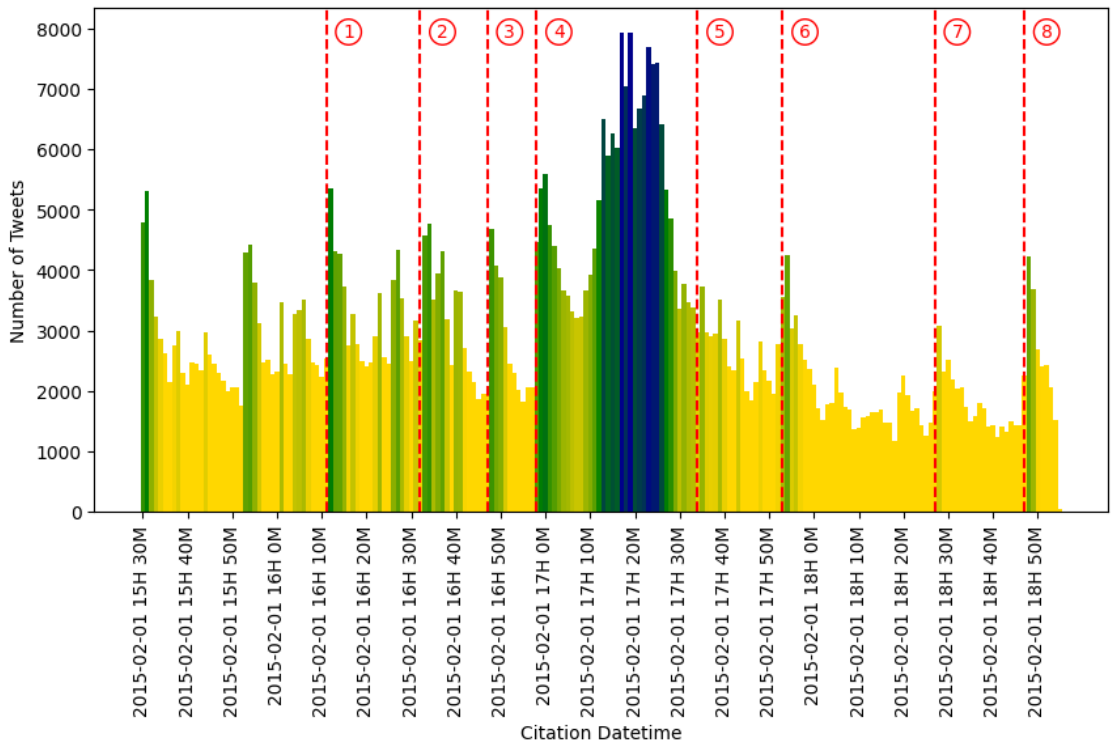
[12]:
```
rootdir = "/tmp2/kch/ece219/ECE219_tweet_data"
dataset = {}

filepaths = get_filepaths(rootdir)
dataset_dict = {filepath: read_file(filepath) for filepath in filepaths}
```

```
0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]
```

[13]:
```
for filepath in filepaths:
    print(filepath)
    df = dataset_dict[filepath]
    df = df[(df['citation_date'] >= game_start) & (df['citation_date'] <=␣
↪game_end)]
    df['citation_date_datetime'] = df['citation_date'].apply(datetime.datetime.
↪fromtimestamp, args=(pst_tz,))
    tweets_per_minute_plot(df,
                           title_prefix=filepath.split("#")[1][:-4])
```

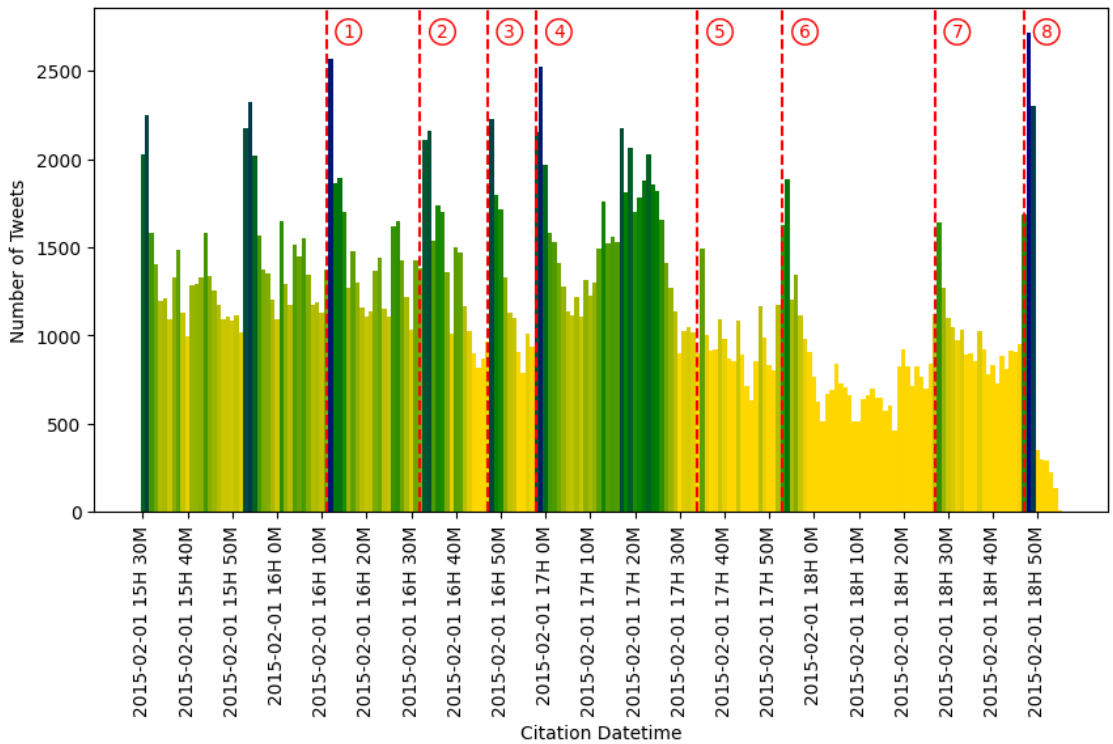
```
/tmp2/kch/ece219/ECE219_tweet_data/tweets_#superbowl.txt

/tmp/ipykernel_27643/2362341121.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
```

```
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```

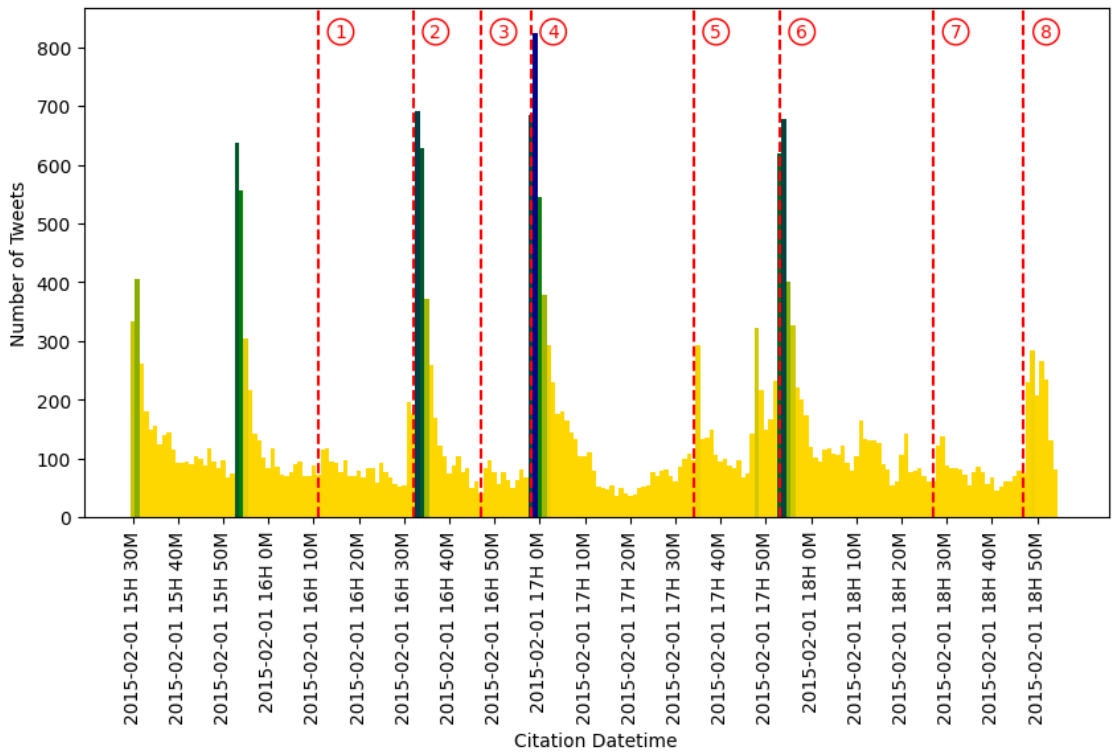



```
/tmp2/kch/ece219/ECE219_tweet_data/tweets_#sb49.txt
```

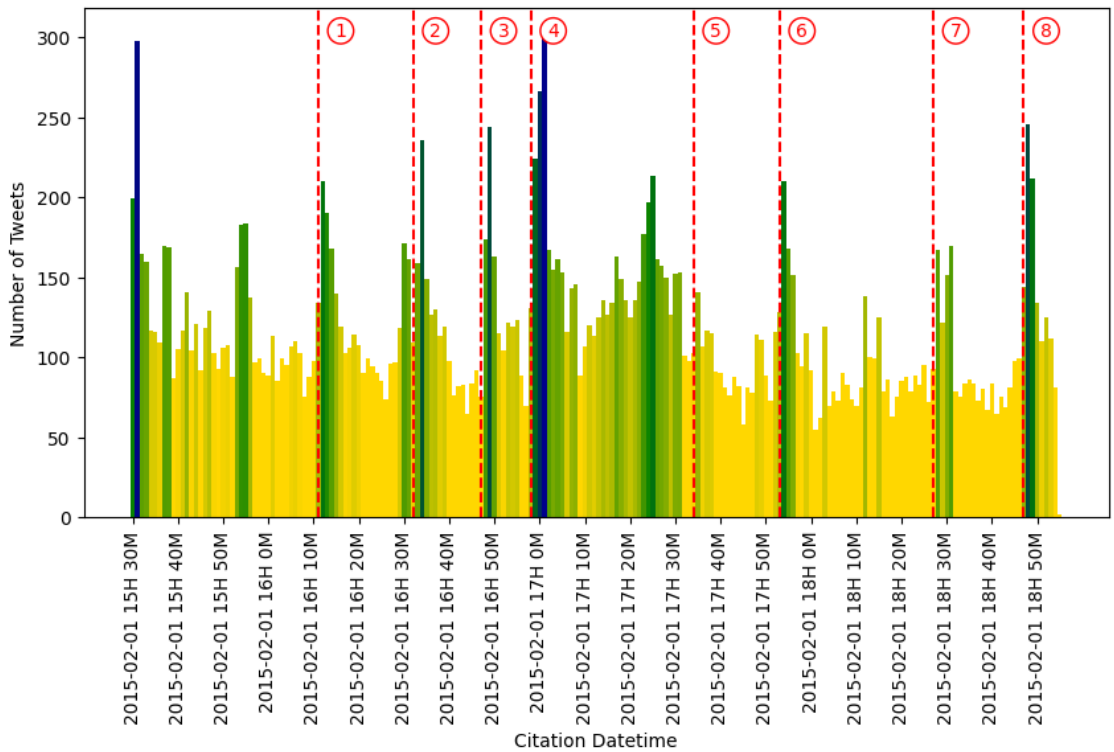
```
/tmp/ipykernel_27643/2362341121.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```



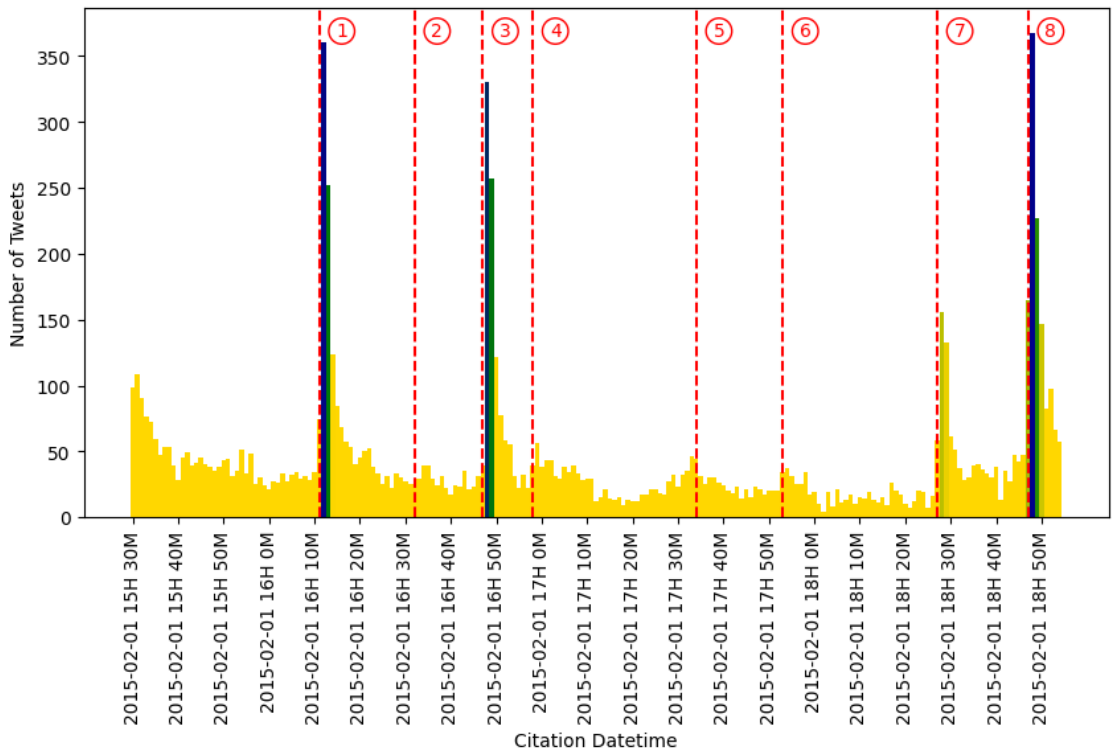

```
/tmp2/kch/ece219/ECE219_tweet_data/tweets_#gohawks.txt

/tmp/ipykernel_27643/2362341121.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```



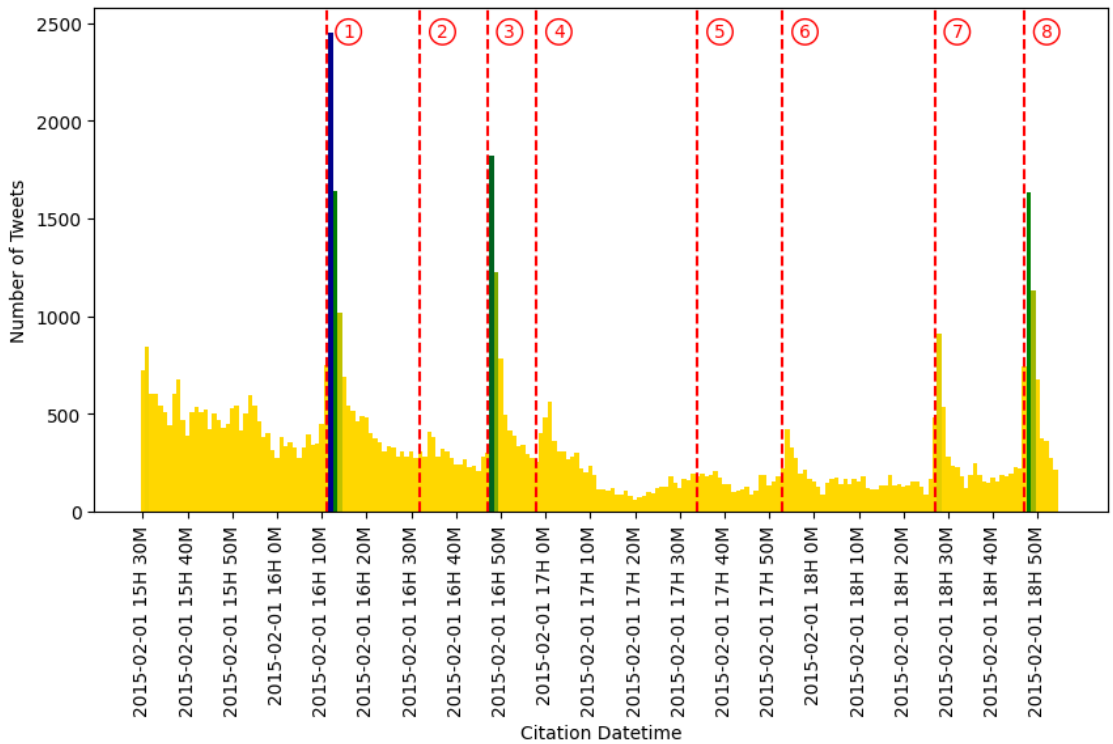

```
/tmp2/kch/ece219/ECE219_tweet_data/tweets_#nfl.txt

/tmp/ipykernel_27643/2362341121.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```




```
/tmp2/kch/ece219/ECE219_tweet_data/tweets_#gopatriots.txt

/tmp/ipykernel_27643/2362341121.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```




```
/tmp2/kch/ece219/ECE219_tweet_data/tweets_#patriots.txt

/tmp/ipykernel_27643/2362341121.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```



[ ]:

```
```

[14]:

```
data_dir = "/tmp2/kch/ece219/ECE219_tweet_data"
filename = "tweets_#gohawks.txt"
data = []
n = 100

with open(os.path.join(data_dir, filename)) as f:
    for line in tqdm(f):
        all_info = json.loads(line)
        # print(all_info["tweet"]["user"])
        taken_info = {
#             "type": all_info["type"],
#             # "title": all_info["title"],
            "text": all_info["tweet"]["text"],
            "id": all_info["tweet"]["id"],
#             "hashtags": " ".join([
#                 " ".join([s for s in re.split('([A-Z][a-z]+)', h["text"]) if␣
 ↪s])
#                 for h in all_info["tweet"]["entities"]["hashtags"]
#             ]),  # hashtag split
            "location": all_info["tweet"]["user"]['location'],
#             "citation_date": all_info["citation_date"],
```

```
        }
        data.append(taken_info)

data_df = pd.DataFrame(data)
```

```
0it [00:00, ?it/s]
```

[15]:
```
data_df
```

[15]:
```
                                                   text                  id  \
0       I &lt;3 our defense! #GoHawks http://t.co/U1pc…  549327579782840320
1       twelfth dogs are ready! #gohawks #dogslife htt…  549575600210718721
2       "Oh no big deal, just NFC West Champs and the …  549647876406534144
3       At http://t.co/VdORWOeAed -- #Seahawks #12thMA…  549658771749101568
4       Good luck at Michigan, Jim Harbaugh. #GoHawks …  549989186950803456
…                                                     …                   …
169117  How long to you think that it will be until we…  563946446702264320
169118  #Seahawks all day, everyday!!! #beastmode #LOB…  563952234178691072
169119  Interview With the Seattle Seahawks Superbowl …  563968915793903617
169120  If you love Russell Wilson and want to know mo…  563969561184460800
169121  The ultimate broken hearted hangover week from…  564005099874181122

                location
0                Ontario
1            Redmond, WA
2                Seattle
3
4       Buffalo, Wyoming
…                     …
169117           Seattle
169118          So. Cal,
169119        Washington
169120           Seattle
169121       Seattle, WA

[169122 rows x 3 columns]
```

[16]:
```
from geopy.exc import GeocoderTimedOut
from geopy.geocoders import Nominatim

from tqdm.auto import tqdm

# Used to map the twitter user location description to a standard format
def lookup_location(location, attempt=1, max_attempts=5):
    geo_locator = Nominatim(user_agent="LearnPython", timeout=None)
    try:
        location = geo_locator.geocode(location, language='en')
```

```
    except GeocoderTimedOut:
        if attempt <= max_attempts:
            return lookup_location(location, attempt=attempt+1)
        raise
    except:
        return
    return location
```

[17]:
```
# data_df['map_location'] = data_df['location'].apply(lookup_location)
```

[18]:
```
map_location = []
n = 0
```

[19]:
```
try:
    for loc in tqdm(data_df['location']):
        # print(loc)
        map_location.append(lookup_location(loc))
        n += 1
        if n == 33000:
            break
except KeyboardInterrupt:
    pass
```

```
  0%|          | 0/169122 [00:00<?, ?it/s]
```

[20]:
```
map_location
```

[20]:
```
[Location(Ontario, Canada, (50.000678, -86.000977, 0.0)),
 Location(Redmond, King County, Washington, United States, (47.6694141,
-122.1238767, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Buffalo, Johnson County, Wyoming, 82834, United States, (44.3480865,
-106.699287, 0.0)),
 Location(McChord Air Force Base, Fairway Road, Oakswest Apartments, Pierce
County, Washington, 98438, United States, (47.1309548, -122.502686, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 None,
 Location(Pacific Northwest College of Art - Schnitzer Center, 511, Northwest
Broadway, Pearl District, Portland, Multnomah County, Oregon, 97209, United
States, (45.5270896, -122.678128, 0.0)),
 None,
 None,
 None,
 None,
 None,
```

```
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 None,
 None,
 None,
 Location(Bellevue, King County, Washington, United States, (47.6144219,
-122.192337, 0.0)),
 Location(Kawilihan Village, Pasig, Eastern Manila District, Metro Manila,
Philippines, (14.567318799999999, 121.06816191904761, 0.0)),
 Location(Las Vegas, Clark County, Nevada, United States, (36.1672559,
-115.148516, 0.0)),
 None,
 None,
 Location(PNW Trail, Samish, Skagit County, Washington, 98229, United States,
(48.614983, -122.419388, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Shelton, Mason County, Washington, United States, (47.2150945,
-123.100706, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Edmonds, Snohomish County, Washington, United States, (47.8105738,
-122.3774952, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Greater Friendship Church, 2623, East Union Street, Madison Valley,
Central District, Seattle, King County, Washington, 98122, United States,
(47.6127397, -122.29771650051924, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Spokane, Spokane County, Washington, United States, (47.6571934,
-117.42351, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Camp Murray, Pierce County, Washington, 98430, United States,
(47.1174101, -122.56813863470596, 0.0)),
 None,
 Location(Monroe, Snohomish County, Washington, 98272, United States,
(47.8552259, -121.971335, 0.0)),
 Location(T-Mobile Park, South Royal Brougham Way, International
District/Chinatown, Seattle, King County, Washington, 98104, United States,
(47.59122645, -122.33256510340122, 0.0)),
```

```
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Bellingham, Whatcom County, Washington, United States, (48.7544012,
-122.478836, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 None,
 Location(Issaquah, King County, Washington, United States, (47.5348778,
-122.0432974, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(SeaTac, King County, Washington, United States, (47.4449086,
-122.283084, 0.0)),
 Location(Auburn, King County, Washington, United States, (47.3075369,
-122.2301808, 0.0)),
 None,
 Location(Australia Avenue, Broadbeach, Gold Coast City, Queensland, 4218,
Australia, (-28.0246204, 153.4321536, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Issaquah, King County, Washington, United States, (47.5348778,
-122.0432974, 0.0)),
 Location(Bellevue, King County, Washington, United States, (47.6144219,
-122.192337, 0.0)),
 Location(Pier 69, Alaskan Way, West Edge, Belltown, Seattle, King County,
Washington, 98121, United States, (47.614021699999995, -122.35468520146657,
0.0)),
 Location(Renton, King County, Washington, United States, (47.4799078,
-122.2034496, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(T-Mobile Park, South Royal Brougham Way, International
District/Chinatown, Seattle, King County, Washington, 98104, United States,
```

```
(47.59122645, -122.33256510340122, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Edgewood, Clayton County, Iowa, United States, (42.6444914,
-91.4011118, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Florida, United States, (27.7567667, -81.4639835, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(San Diego, San Diego County, California, United States, (32.7174202,
-117.1627728, 0.0)),
 None,
 Location(SMS, Kurtovići, Klis, Općina Klis, Split-Dalmatia County, Croatia,
(43.57646, 16.553003927047587, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Canton, Stark County, Ohio, United States, (40.7985464, -81.3749508,
0.0)),
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(The Town, Bryher, New Grimsby, Isles of Scilly, England, TR23 0PR,
United Kingdom, (49.9545977, -6.3522296, 0.0)),
 Location(Clearwater, Pinellas County, Florida, United States, (27.9658533,
```

```
-82.8001026, 0.0)),
 None,
 Location(San Diego, San Diego County, California, United States, (32.7174202,
-117.1627728, 0.0)),
 Location(Hollywood Erotic Boutique, Lake City Way Northeast, Cedar Park, Lake
City, Seattle, King County, Washington, 98165, United States, (47.7213636,
-122.2936583, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Clearwater, Pinellas County, Florida, United States, (27.9658533,
-82.8001026, 0.0)),
 None,
 Location(Shelton, Mason County, Washington, United States, (47.2150945,
-123.100706, 0.0)),
 None,
 None,
 None,
 Location(Lübeck Airport, Seekrug, Hornstorf, Groß Sarau, Lauenburgische Seen,
Herzogtum Lauenburg, Schleswig-Holstein, 23627, Germany, (53.804942999999994,
10.712610058244511, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Nashville-Davidson, Davidson County, Middle Tennessee, Tennessee,
United States, (36.1622767, -86.7742984, 0.0)),
 Location(Baybay, Bulacan, Central Luzon, 3012, Philippines, (14.94237,
121.0380329, 0.0)),
 None,
 None,
 Location(Tyne and Wear, North East, England, NE9 7YE, United Kingdom,
(54.916667, -1.566667, 0.0)),
 Location(Issaquah, King County, Washington, United States, (47.5348778,
-122.0432974, 0.0)),
 Location(Vancouver, Metro Vancouver Regional District, British Columbia,
Canada, (49.2608724, -123.113952, 0.0)),
 Location(Green Lake County, Wisconsin, United States, (43.7971612, -89.0402508,
0.0)),
 Location(Paradise, Clark County, Nevada, United States, (36.1150858,
-115.173414, 0.0)),
 None,
 Location(Kawilihan Village, Pasig, Eastern Manila District, Metro Manila,
Philippines, (14.567318799999999, 121.06816191904761, 0.0)),
 None,
 None,
```

```
 None,
 Location(Wenatchee, Chelan County, Washington, 98801, United States,
(47.4234599, -120.310349, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Snohomish County, Washington, United States, (48.0074736,
-121.7304882, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(  ,         ,       , Langarud County, Gilan Province, Iran,
(37.0412584, 50.0068157, 0.0)),
 Location(Bellevue, King County, Washington, United States, (47.6144219,
-122.192337, 0.0)),
 None,
 None,
 Location(Iowa City, Johnson County, Iowa, United States, (41.6612561,
-91.5299106, 0.0)),
 Location(Beechers Handmade Cheese, 1600, Pike Place, West Edge, Belltown,
Seattle, King County, Washington, 98101, United States, (47.6095818,
-122.3418158, 0.0)),
 Location(Beverly Hills, Los Angeles County, California, United States,
(34.0696501, -118.3963062, 0.0)),
 None,
 None,
 None,
 None,
 None,
 Location(Green Lake County, Wisconsin, United States, (43.7971612, -89.0402508,
0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Iowa City, Johnson County, Iowa, United States, (41.6612561,
```

```
-91.5299106, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Bend, Deschutes County, Oregon, United States, (44.0581728,
-121.3153096, 0.0)),
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Rio de Janeiro, Região Geográfica Imediata do Rio de Janeiro, Região
Metropolitana do Rio de Janeiro, Região Geográfica Intermediária do Rio de
Janeiro, Rio de Janeiro, Southeast Region, Brazil, (-22.9110137, -43.2093727,
0.0)),
 Location(Red Oak, Cleburne, Johnson County, Texas, 76033, United States,
(32.331846, -97.42331, 0.0)),
 Location(Big Island Glass Gallery, Mamane Street, Haina, Honokaa, Hawai i
County, Hawaii, 96727, United States, (20.0795269, -155.4692986, 0.0)),
 Location(Wisconsin, United States, (44.4308975, -89.6884637, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Oceanography Teaching Building, 3613, San Juan Road Northeast, South
Campus, University District, Seattle, King County, Washington, 98195, United
States, (47.6502531, -122.31230453474096, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Atlanta, Fulton County, Georgia, United States, (33.7489924,
-84.3902644, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Kirkland, King County, Washington, United States, (47.6859573,
-122.1920249, 0.0)),
 None,
```

```
 None,
 Location(Green Lake County, Wisconsin, United States, (43.7971612, -89.0402508,
0.0)),
 Location(Alexandria, Virginia, United States, (38.8051095, -77.0470229, 0.0)),
 None,
 Location(Nance Lakes (Bottom Lake), Ludgvan, Halsetown, St. Ives, Cornwall,
England, United Kingdom, (50.183833699999994, -5.480523625776401, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Spring Green, Sauk County, Wisconsin, United States, (43.1769367,
-90.0680478, 0.0)),
 Location(Cleveland, Cuyahoga County, Ohio, United States, (41.4996574,
-81.6936772, 0.0)),
 Location(Iowa City, Johnson County, Iowa, United States, (41.6612561,
-91.5299106, 0.0)),
 Location(Kirkland, King County, Washington, United States, (47.6859573,
-122.1920249, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(University of Wisconsin-Whitewater, 800, West Main Street, Historic
Starin Park, Whitewater, Walworth County, Wisconsin, 53190, United States,
(42.84435955, -88.74366642884807, 0.0)),
 Location(Pasco, Franklin County, Washington, 99301, United States, (46.2306739,
-119.0921, 0.0)),
 Location(Iowa, United States, (41.9216734, -93.3122705, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(1212, Alvdal, Innlandet, Norway, (62.16212, 10.65908, 0.0)),
 Location(Kirkland, King County, Washington, United States, (47.6859573,
-122.1920249, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Your Sweet Home, 6769, Fremlin Street, Oakridge, Vancouver, Metro
Vancouver Regional District, British Columbia, Canada, (49.224214950000004,
-123.12715595, 0.0)),
 Location(Alaska, United States, (64.4459613, -149.680909, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Aranda de Duero, Burgos, Castile and León, Spain, (41.6715067,
-3.6851172, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
```

```
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(West Seattle, Seattle, King County, Washington, 98116, United States,
(47.5709315, -122.3865166, 0.0)),
 Location(Quantico, Prince William County, Virginia, United States, (38.5218802,
-77.2911023, 0.0)),
 None,
 None,
 Location(West Seattle, Seattle, King County, Washington, 98116, United States,
(47.5709315, -122.3865166, 0.0)),
 Location(West Seattle, Seattle, King County, Washington, 98116, United States,
(47.5709315, -122.3865166, 0.0)),
 Location(Xiqu District, Panzhihua City, Sichuan, China, (26.6031069, 101.62832,
0.0)),
 Location(West Seattle, Seattle, King County, Washington, 98116, United States,
(47.5709315, -122.3865166, 0.0)),
 None,
 Location(Mountlake Terrace, Snohomish County, Washington, 98043, United States,
(47.7909667, -122.3066395, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Augusta, Richmond County, Georgia, United States, (33.4709714,
-81.9748429, 0.0)),
 Location(New York, United States, (40.7127281, -74.0060152, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Las Vegas, Clark County, Nevada, United States, (36.1672559,
-115.148516, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 None,
```

```
 Location(Hinckley, Hinckley and Bosworth, Leicestershire, England, LE10 1NR,
United Kingdom, (52.5410591, -1.3729379, 0.0)),
 None,
 None,
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(City of Saginaw, Saginaw County, Michigan, United States, (43.4200387,
-83.9490365, 0.0)),
 Location(Melbourne, City of Melbourne, Victoria, Australia, (-37.8142176,
144.9631608, 0.0)),
 Location(Wheatland, Oklahoma City, Oklahoma County, Oklahoma, 73097, United
States, (35.3978377, -97.652263, 0.0)),
 None,
 None,
 Location(Panama, (8.559559, -81.1308434, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Decatur County, Georgia, United States, (30.8736613, -84.5740953,
0.0)),
 Location(McLeod, Sandoun Township, Ransom County, North Dakota, United States,
(46.3935753, -97.2995351, 0.0)),
 Location(Leeds Close, Southwater, Horsham, West Sussex, England, RH13 9XN,
United Kingdom, (51.0267683, -0.3432113, 0.0)),
 None,
 Location(New Orleans, Orleans Parish, Louisiana, United States, (29.9759983,
-90.0782127, 0.0)),
 Location(Atlanta, Fulton County, Georgia, United States, (33.7489924,
-84.3902644, 0.0)),
 Location(Atlanta, Fulton County, Georgia, United States, (33.7489924,
-84.3902644, 0.0)),
 None,
 Location(Maryland, United States, (39.5162401, -76.9382069, 0.0)),
 None,
 None,
 None,
 None,
 Location(Nottingham, Nottinghamshire, England, United Kingdom, (52.9534193,
-1.1496461, 0.0)),
 Location(Boston, Suffolk County, Massachusetts, United States, (42.3554334,
-71.060511, 0.0)),
 Location(New York, United States, (40.7127281, -74.0060152, 0.0)),
 Location(Boston, Suffolk County, Massachusetts, United States, (42.3554334,
-71.060511, 0.0)),
 Location(Museum of the Great Plains, 601, Northwest Ferris Avenue, Lawton,
Comanche County, Oklahoma, 73507, United States, (34.6178934,
-98.39812831435282, 0.0)),
```

```
 Location(Kiel, Schleswig-Holstein, Germany, (54.3227085, 10.135555, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Athens-Clarke County Unified Government, Athens-Clarke County,
Georgia, United States, (33.9597677, -83.376398, 0.0)),
 Location(Atlanta, Fulton County, Georgia, United States, (33.7489924,
-84.3902644, 0.0)),
 Location(Minneapolis, Hennepin County, Minnesota, United States, (44.9772995,
-93.2654692, 0.0)),
 Location(Kansas City, Jackson County, Missouri, United States, (39.100105,
-94.5781416, 0.0)),
 Location(Kansas City, Jackson County, Missouri, United States, (39.100105,
-94.5781416, 0.0)),
 Location(Florida, United States, (27.7567667, -81.4639835, 0.0)),
 Location(Florida, United States, (27.7567667, -81.4639835, 0.0)),
 None,
 Location(Florida, United States, (27.7567667, -81.4639835, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Florida, United States, (27.7567667, -81.4639835, 0.0)),
 Location(Whidbey Island, Island County, Washington, United States,
(48.15814255, -122.59742939440014, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 None,
 None,
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Spanaway, Pierce County, Washington, United States, (47.0979462,
-122.4243422, 0.0)),
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
```

```
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 Location(Kirkland, King County, Washington, United States, (47.6859573,
-122.1920249, 0.0)),
 Location(Blaine, Whatcom County, Washington, 98231, United States, (48.9881108,
-122.743575, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(Lelystad, Flevoland, Netherlands, (52.5150949, 5.4768915, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 None,
 Location(Heath Hawk Drive, Heath, Rockwall County, Texas, 75032, United States,
(32.8442426, -96.4776166, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(Atlanta, Madibeng Ward 14, Madibeng Local Municipality, Bojanala
Platinum District Municipality, North West, South Africa, (-25.249762,
27.6142922, 0.0)),
 Location(Vancouver, Metro Vancouver Regional District, British Columbia,
Canada, (49.2608724, -123.113952, 0.0)),
 Location(Las Vegas, Comuna 12 - Ricaurte, Perímetro Urbano Ibagué, Ibagué,
Centro, Tolima, 730001, Colombia, (4.4332214, -75.2374331, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 None,
 Location(Spanaway, Pierce County, Washington, United States, (47.0979462,
-122.4243422, 0.0)),
```

```
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Ohio, United States, (40.2253569, -82.6881395, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Bellevue, King County, Washington, United States, (47.6144219,
-122.192337, 0.0)),
 None,
 Location(5116, Perry Drive Southeast, Lakeland Hills, Auburn, King County,
Washington, 98092, United States, (47.2628229, -122.20970847155847, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 None,
 None,
 Location(Brisbane City, Queensland, Australia, (-27.4689682, 153.0234991,
0.0)),
 None,
 None,
 Location(Madrid, Área metropolitana de Madrid y Corredor del Henares, Community
of Madrid, Spain, (40.4167047, -3.7035825, 0.0)),
 None,
 None,
 None,
 Location(Vancouver, Clark County, Washington, United States, (45.6306954,
-122.6744557, 0.0)),
 Location(Madrid, Área metropolitana de Madrid y Corredor del Henares, Community
of Madrid, Spain, (40.4167047, -3.7035825, 0.0)),
 None,
 None,
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Mercer Island, King County, Washington, 98040, United States,
(47.5602073, -122.22014226861951, 0.0)),
 None,
 Location(Lake Stevens, Snohomish County, Washington, 98258, United States,
```

```
(48.0197794, -122.0660914, 0.0)),
 None,
 Location(West Seattle, Seattle, King County, Washington, 98116, United States,
(47.5709315, -122.3865166, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Charlotte, Mecklenburg County, North Carolina, United States,
(35.2272086, -80.8430827, 0.0)),
 None,
 Location(Montana, United States, (47.3752671, -109.638757, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 Location(Church at South Seattle, 3019, South Angeline Street, Rainier Vista,
Rainier Valley, Seattle, King County, Washington, 98108, United States,
(47.5590962, -122.2927589, 0.0)),
 None,
 Location(DF, Texcoco de Mora, Texcoco, State of Mexico, 56130, Mexico,
(19.5121601, -98.8743728, 0.0)),
 None,
 Location(Olympia, Thurston County, Washington, United States, (47.0451022,
-122.8950075, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(Fresno, Fresno County, California, United States, (36.7394421,
-119.7848307, 0.0)),
 Location(Marysville, Snohomish County, Washington, United States, (48.0517429,
-122.1768209, 0.0)),
 Location(Plainview, Hale County, 79072, United States, (34.1847936,
-101.7068417, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(New York, United States, (40.7127281, -74.0060152, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
```

```
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 None,
 None,
 Location(Maple Valley, King County, Washington, 98038, United States,
(47.3664231, -122.0437127, 0.0)),
 None,
 Location(Haleyville, Winston County, Alabama, 35565, United States, (34.226488,
-87.6214133, 0.0)),
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Bend, Deschutes County, Oregon, United States, (44.0581728,
-121.3153096, 0.0)),
 Location(Big Valley, Henry Parkes Way, The Dungeons, Parkes, Parkes Shire
Council, New South Wales, 2870, Australia, (-33.1468136, 148.3222440951613,
0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Ottawa, Eastern Ontario, Ontario, Canada, (45.4208777, -75.6901106,
0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Lake Stevens, Snohomish County, Washington, 98258, United States,
(48.0197794, -122.0660914, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 None,
 None,
 None,
 None,
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Atlanta, Fulton County, Georgia, United States, (33.7489924,
-84.3902644, 0.0)),
 Location(Minnesota, United States, (45.9896587, -94.6113288, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 None,
```

```
 Location(Renton, King County, Washington, United States, (47.4799078,
-122.2034496, 0.0)),
 None,
 Location(Berlin, Germany, (52.5170365, 13.3888599, 0.0)),
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(United States, (39.7837304, -100.445882, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 None,
 None,
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Oregon, Municipality of Victoria County, Victoria County, Nova Scotia,
Canada, (46.3187056, -60.6411688, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Puyallup, Pierce County, Washington, United States, (47.1849009,
-122.29214, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(New England, United States, (44.2057083, -70.75378387447887, 0.0)),
 Location(Spokane, Spokane County, Washington, United States, (47.6571934,
-117.42351, 0.0)),
 Location(Moving, Via Fiorentina, La Cella, Putignano, Ospedaletto, Pisa,
Tuscany, 56124, Italy, (43.700902, 10.4208382, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Success Farm to Spot Valley Road, Spot Valley, Saint James, Cornwall
County, Jamaica, (18.4964805, -77.7912953, 0.0)),
 Location(Iowa City, Johnson County, Iowa, United States, (41.6612561,
```

-91.5299106, 0.0)),
 Location(Small Town, Kertamukti, Garut, West Java, Indonesia, (-7.5036904, 107.70552178121659, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
 Location(Raleigh, Wake County, North Carolina, United States, (35.7803977, -78.6390989, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013, -122.438329, 0.0)),
 Location(Dallas/Fort Worth International Airport, Grapevine, Tarrant County, 75261, United States, (32.89651945, -97.0465220537124, 0.0)),
 Location(Big Valley, Henry Parkes Way, The Dungeons, Parkes, Parkes Shire Council, New South Wales, 2870, Australia, (-33.1468136, 148.3222440951613, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
 Location(Yakima, Yakima County, Washington, United States, (46.601557, -120.510842, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 Location(Bonney Lake, Pierce County, Washington, United States, (47.1870191, -122.1702293, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013, -122.438329, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
 Location(Olympia, Π E O 74, Ancient Olympia, Municipality of Ancient Olympia, Elis Regional Unit, Western Greece, Peloponnese, Western Greece and the Ionian, 270 65, Greece, (37.638250299999996, 21.630566024011667, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013, -122.438329, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368, -77.0365427, 0.0)),
 Location(Jamaica, (18.1850507, -77.3947693, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013, -122.438329, 0.0)),
 None,

Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(South Puget Sound Community College - Lacey Campus, Golf Club Road Southeast, Lacey, Thurston County, Washington, 98503, United States, (47.044289199999994, -122.8297792928368, 0.0)),
Location(Baso Ony, Jalan Insinyur Haji Juanda, Cipedes, Tasikmalaya, West Java, 46151, Indonesia, (-7.305449, 108.2043531, 0.0)),
None,
None,
None,
Location(Seattle Heights, Lynnwood, Snohomish County, Washington, 98026, United States, (47.8103747, -122.32402, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
None,
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
None,
Location(Bend, Deschutes County, Oregon, United States, (44.0581728, -121.3153096, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
None,
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
None,
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(Chicago, Cook County, Illinois, United States, (41.8755616, -87.6244212, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
None,
None,
Location(Zone6, Sint-Antoniusweg, Shipit, Kallo, Beveren, Sint-Niklaas, East Flanders, Flanders, 9130, Belgium, (51.2750357, 4.2477275, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
None,
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(North Rhine-Westphalia, Germany, (51.4789205, 7.5543751, 0.0)),
Location(Mahomet, Champaign County, Illinois, 61853, United States, (40.195142,

-88.4052266, 0.0)),
 None,
 Location(Des Moines, Polk County, Iowa, United States, (41.5910323,
-93.6046655, 0.0)),
 None,
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 None,
 None,
 None,
 None,
 None,
 None,
 Location(Ooltewah, Hamilton County, East Tennessee, Tennessee, United States,
(35.072036, -85.08885976005197, 0.0)),
 Location(Kirkland, King County, Washington, United States, (47.6859573,
-122.1920249, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(Victoria, Capital Regional District, British Columbia, Canada,
(48.4283182, -123.3649533, 0.0)),
 Location(Puget Sound, King County, Washington, United States, (47.7385375,
-122.43027529817088, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Vancouver, Metro Vancouver Regional District, British Columbia,
Canada, (49.2608724, -123.113952, 0.0)),
 Location(Brem Rock Pit, 1st Street, West Park, Bremerton, Kitsap County,
Washington, 98312, United States, (47.5642, -122.6833, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(Shoreline, King County, Washington, 98155, United States, (47.7564667,
-122.3437497, 0.0)),
 None,
 Location(Cheney, Spokane County, Washington, United States, (47.4892246,
-117.579304, 0.0)),
 Location(Cheney, Avallon, Yonne, Bourgogne-Franche-Comté, Metropolitan France,
89700, France, (47.9073765, 3.9477501, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,

```
 Location(201, King Street, North Kamloops, Kamloops, Thompson-Nicola Regional
District, British Columbia, V2B 3Z8, Canada, (50.6896702, -120.3616734, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 None,
 None,
 Location(919, Spruce Pine Trail, Whispering Pines, Durham, Orange County, North
Carolina, 27705, United States, (36.031385, -79.0076104, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Central, Cleveland, Cuyahoga County, Ohio, 44101, United States,
(41.4927839, -81.657708, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Vancouver, Metro Vancouver Regional District, British Columbia,
Canada, (49.2608724, -123.113952, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Olympia, Thurston County, Washington, United States, (47.0451022,
-122.8950075, 0.0)),
 None,
 None,
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Spanaway, Pierce County, Washington, United States, (47.0979462,
```

-122.4243422, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Muscatine County, Iowa, United States, (41.4776121, -91.1210053,
0.0)),
 None,
 Location(Nashville-Davidson, Davidson County, Middle Tennessee, Tennessee,
United States, (36.1622767, -86.7742984, 0.0)),
 None,
 Location(West Palm Beach, Palm Beach County, Florida, United States,
(26.715364, -80.0532942, 0.0)),
 Location(Jarvis, Haldimand County, Ontario, N0A 1J0, Canada, (42.8844706,
-80.1139414, 0.0)),
 None,
 Location(Bonney Lake, Pierce County, Washington, United States, (47.1870191,
-122.1702293, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(North Carolina, United States, (35.6729639, -79.0392919, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Lou's City Bar, 1400, Irving Street Northwest, Columbia Heights,
Washington, District of Columbia, 20010, United States, (38.9284992, -77.033826,
0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Spokane, Spokane County, Washington, United States, (47.6571934,
-117.42351, 0.0)),
 None,
 Location(Spokane Valley, Spokane County, Washington, United States,
(47.6571104, -117.2613936, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Embassy of Hungary, 1500, Rhode Island Avenue Northwest, Dupont
Circle, Washington, District of Columbia, 20005, United States, (38.9074322,
-77.0350922, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Anacortes, Skagit County, Washington, 98221, United States,

```
(48.5020123, -122.623735, 0.0)),
 Location(Kirkland, King County, Washington, United States, (47.6859573,
-122.1920249, 0.0)),
 Location(Georgia, (41.6809707, 44.0287382, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Toluca Lake, Los Angeles, Los Angeles County, California, 91602,
United States, (34.1521688, -118.357141, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Capitol Hill, Seattle, King County, Washington, 98102, United States,
(47.6238307, -122.318368, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 None,
 None,
 None,
 None,
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Regina International Airport, 5210, Regina Avenue, River Heights,
Regina, Division No. 6, Saskatchewan, S4W 1B3, Canada, (50.4309977,
-104.66560598444946, 0.0)),
 Location(Touchdown, Mendoza, Barrio Chino, Belgrano, Buenos Aires, Comuna 13,
Autonomous City of Buenos Aires, C1428DKK, Argentina, (-34.5609942, -58.4572061,
0.0)),
 None,
 Location(State of Mind, 34, Kolmas linja, Linjat, Kallio, Central major
district, Helsinki, Helsinki sub-region, Uusimaa, Southern Finland, Mainland
Finland, 00530, Finland, (60.1849017, 24.9425014, 0.0)),
 Location(Tent City Jail, 2939, West Durango Street, Phoenix, Maricopa County,
Arizona, 85009, United States, (33.42767815, -112.12409514412093, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 Location(Woodinville, King County, Washington, United States, (47.7545827,
-122.1588902, 0.0)),
```

None,
Location(New York, United States, (40.7127281, -74.0060152, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
None,
Location(Vancouver, Clark County, Washington, United States, (45.6306954,
-122.6744557, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
Location(Salt Lake City, Salt Lake County, Utah, United States, (40.7596198,
-111.886797, 0.0)),
Location(Anderson Island, Pierce County, Washington, 98303, United States,
(47.1557835, -122.70709522798506, 0.0)),
Location(Kirkland, King County, Washington, United States, (47.6859573,
-122.1920249, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
None,
None,
Location(407, Route 253, Eaton Corner, Eaton, Cookshire-Eaton, Le Haut-Saint-
François, Estrie, Quebec, J0B 3A0, Canada, (45.3676823, -71.5985261, 0.0)),
None,
Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
Location(Palatine, Palatine Township, Cook County, Illinois, United States,
(42.1105779, -88.0336888, 0.0)),
None,
Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
None,
Location(1402, Southeast Everett Mall Way, Twin Creeks, Everett, Snohomish
County, Washington, 98208, United States, (47.911984491971744,
-122.21427638422014, 0.0)),
Location(Federal Way, King County, Washington, United States, (47.313494,
-122.33931, 0.0)),
Location(916, Sacramento Avenue, Broderick, West Sacramento, Yolo County,
California, 95605, United States, (38.590067, -121.5247711, 0.0)),
None,
Location(United States, (39.7837304, -100.445882, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
Location(Tacoma, Pierce County, Washington, United States, (47.2455013,

```
-122.438329, 0.0)),
 None,
 Location(Old Toronto, Toronto, Golden Horseshoe, Ontario, Canada, (43.6534817,
-79.3839347, 0.0)),
 Location(Old Toronto, Toronto, Golden Horseshoe, Ontario, Canada, (43.6534817,
-79.3839347, 0.0)),
 Location(Old Toronto, Toronto, Golden Horseshoe, Ontario, Canada, (43.6534817,
-79.3839347, 0.0)),
 Location(Old Toronto, Toronto, Golden Horseshoe, Ontario, Canada, (43.6534817,
-79.3839347, 0.0)),
 Location(Old Toronto, Toronto, Golden Horseshoe, Ontario, Canada, (43.6534817,
-79.3839347, 0.0)),
 Location(Old Toronto, Toronto, Golden Horseshoe, Ontario, Canada, (43.6534817,
-79.3839347, 0.0)),
 None,
 Location(Old Toronto, Toronto, Golden Horseshoe, Ontario, Canada, (43.6534817,
-79.3839347, 0.0)),
 None,
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Clark College, James P Brawley Drive Southwest, Atlanta University
Center, Atlanta, Fulton County, Georgia, 30314, United States, (33.7483398,
-84.4119502, 0.0)),
 Location(Boston, Suffolk County, Massachusetts, United States, (42.3554334,
-71.060511, 0.0)),
 Location(Seattle, 2701, Alaskan Way, West Edge, Belltown, Seattle, King County,
Washington, 98121, United States, (47.6134905, -122.35349, 0.0)),
 Location(Colorado Springs, El Paso County, Colorado, United States,
(38.8339578, -104.825348, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(1402, Southeast Everett Mall Way, Twin Creeks, Everett, Snohomish
County, Washington, 98208, United States, (47.911984491971744,
-122.21427638422014, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
```

```
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Native, Convict Lake Road, Mono County, California, United States,
(37.597623049999996, -118.85070124999999, 0.0)),
 Location(Yakima, Yakima County, Washington, United States, (46.601557,
-120.510842, 0.0)),
 None,
 Location(Seattle Lane, The Preserve, Redwood City, San Mateo County,
California, 94065, United States, (37.5453138, -122.2436197, 0.0)),
 Location(Issaquah, King County, Washington, United States, (47.5348778,
-122.0432974, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Renton, King County, Washington, United States, (47.4799078,
-122.2034496, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Jacksonville, Duval County, Florida, United States, (30.3321838,
-81.655651, 0.0)),
 None,
 None,
 Location(Wa State (Northern Region), Myanmar, (22.6237983, 98.972023075, 0.0)),
 Location(Plymouth, England, United Kingdom, (50.3712659, -4.1425658, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Austin, Travis County, Texas, United States, (30.2711286, -97.7436995,
0.0)),
 Location(Here, Prozor-Rama municipality, Herzegovina-Neretva Canton, Federation
of Bosnia and Herzegovina, Bosnia and Herzegovina, (43.8222157, 17.6896937,
0.0)),
 Location(Southend-on-Sea, England, United Kingdom, (51.5388241, 0.7128137,
0.0)),
 None,
```

```
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 None,
 Location(Eatonville, Pierce County, Washington, United States, (46.8682516,
-122.270068, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 None,
 Location(Olympia, Thurston County, Washington, United States, (47.0451022,
-122.8950075, 0.0)),
 Location(Jacksonville, Duval County, Florida, United States, (30.3321838,
-81.655651, 0.0)),
 None,
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Let The Good Times Roll, Knysna Ward 4, Knysna Local Municipality,
Garden Route District Municipality, Western Cape, 6570, South Africa,
(-34.0004292, 23.0401167, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Kent, King County, Washington, United States, (47.3826903,
-122.2270272, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Vancouver, Metro Vancouver Regional District, British Columbia,
Canada, (49.2608724, -123.113952, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
```

```
-122.330062, 0.0)),
 None,
 Location(Alsip, Worth Township, Cook County, Illinois, 60803, United States,
(41.6689223, -87.7386619, 0.0)),
 None,
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 None,
 Location(Puyallup, Pierce County, Washington, United States, (47.1849009,
-122.29214, 0.0)),
 None,
 None,
 None,
 None,
 None,
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Bellingham, Whatcom County, Washington, United States, (48.7544012,
-122.478836, 0.0)),
 Location(Auburn, King County, Washington, United States, (47.3075369,
-122.2301808, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Panama, (8.559559, -81.1308434, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Kitchener, Region of Waterloo, Ontario, Canada, (43.451291,
-80.4927815, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
```

Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(Savannah rapids playground, Savannah Rapid Trail, Columbia County, Georgia, United States, (33.549815100000004, -82.04035723187772, 0.0)),
Location(Yakima, Yakima County, Washington, United States, (46.601557, -120.510842, 0.0)),
Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
Location(Las Vegas, Comuna 12 - Ricaurte, Perímetro Urbano Ibagué, Ibagué, Centro, Tolima, 730001, Colombia, (4.4332214, -75.2374331, 0.0)),
Location(London, Greater London, England, United Kingdom, (51.5073359, -0.12765, 0.0)),
Location(Poulsbo, Kitsap County, Washington, United States, (47.7391366, -122.63928, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(Mill Creek Hall, 73rd Avenue West, Seattle Heights, Lynnwood, Snohomish County, Washington, 98026, United States, (47.8162498, -122.32977055, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(Centre de secours principal de Calais, Rue Louise Michel, Fort-Nieulay, Calais, Pas-de-Calais, Hauts-de-France, Metropolitan France, 62100, France, (50.9495342, 1.8345309749999996, 0.0)),
Location(England, United Kingdom, (52.5310214, -1.2649062, 0.0)),
Location(The West Side, Eureka, Humboldt County, California, 95502, United States, (40.7915961, -124.1749482, 0.0)),
None,
None,
None,
None,
Location(Danbury High School, 43, Clapboard Ridge Road, Jeanette Heights, Danbury, Western Connecticut COG, Connecticut, 06811, United States, (41.4169585, -73.47022266248165, 0.0)),
None,
Location(Tacoma, Pierce County, Washington, United States, (47.2455013, -122.438329, 0.0)),
None,
Location(Tacoma, Pierce County, Washington, United States, (47.2455013, -122.438329, 0.0)),
None,
None,
None,
Location(Seattle, King County, Washington, United States, (47.6038321, -122.330062, 0.0)),
Location(Tacoma, Pierce County, Washington, United States, (47.2455013, -122.438329, 0.0)),
Location(Seattle, King County, Washington, United States, (47.6038321,

```
-122.330062, 0.0)),
 Location(Calgary, Alberta, Canada, (51.0460954, -114.065465, 0.0)),
 None,
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Mill Creek Hall, 73rd Avenue West, Seattle Heights, Lynnwood,
Snohomish County, Washington, 98026, United States, (47.8162498, -122.32977055,
0.0)),
 Location(Kent, King County, Washington, United States, (47.3826903,
-122.2270272, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Federal Way, King County, Washington, United States, (47.313494,
-122.33931, 0.0)),
 Location(Alma Chocolate, Northeast Davis Street, Kerns, Portland, Multnomah
County, Oregon, 97232, United States, (45.52418375, -122.63708107498121, 0.0)),
 Location(Baso Ony, Jalan Insinyur Haji Juanda, Cipedes, Tasikmalaya, West Java,
46151, Indonesia, (-7.305449, 108.2043531, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 None,
 None,
 None,
 None,
 None,
 Location(Atlanta, Fulton County, Georgia, United States, (33.7489924,
-84.3902644, 0.0)),
 None,
```

```
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Spokane, Spokane County, Washington, United States, (47.6571934,
-117.42351, 0.0)),
 Location(1212, Alvdal, Innlandet, Norway, (62.16212, 10.65908, 0.0)),
 Location(Iowa City Terminal, Coralville, Johnson County, Iowa, United States,
(41.68120825, -91.57071454245758, 0.0)),
 Location(1212, Alvdal, Innlandet, Norway, (62.16212, 10.65908, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Las Vegas, Comuna 12 - Ricaurte, Perímetro Urbano Ibagué, Ibagué,
Centro, Tolima, 730001, Colombia, (4.4332214, -75.2374331, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Wisconsin Rapids, Wood County, Wisconsin, United States, (44.3917589,
-89.8228767, 0.0)),
 Location(Lost, Bellabeg, Aberdeenshire, Scotland, AB36 8UQ, United Kingdom,
(57.2053788, -3.0793008, 0.0)),
 Location(Kent, King County, Washington, United States, (47.3826903,
-122.2270272, 0.0)),
 Location(Camano Island, Island County, Washington, 98282, United States,
(48.1625483, -122.49502681973634, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Old Toronto, Toronto, Golden Horseshoe, Ontario, Canada, (43.6534817,
-79.3839347, 0.0)),
 Location(Santa Barbara, Santa Barbara County, California, United States,
(34.4221319, -119.702667, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 Location(Seattle House Apartments, Delaware, Delaware County, Ohio, United
States, (40.2961181, -83.03122937024887, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Iowa City, Johnson County, Iowa, United States, (41.6612561,
-91.5299106, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 Location(Ontario, Canada, (50.000678, -86.000977, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
```

```
-122.330062, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Pacific Northwest College of Art - Schnitzer Center, 511, Northwest
Broadway, Pearl District, Portland, Multnomah County, Oregon, 97209, United
States, (45.5270896, -122.678128, 0.0)),
 None,
 None,
 Location(Yakima, Yakima County, Washington, United States, (46.601557,
-120.510842, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(PNW Trail, Samish, Skagit County, Washington, 98229, United States,
(48.614983, -122.419388, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 Location(Kenmore, King County, Washington, 98028, United States, (47.7573202,
-122.2440148, 0.0)),
 Location(Roseville, Placer County, California, United States, (38.7521235,
-121.2880059, 0.0)),
 Location(Port Orchard, Kitsap County, Washington, United States, (47.5315625,
-122.6384056, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 None,
 None,
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Port Angeles, Clallam County, Washington, 98362, United States,
(48.118146, -123.430741, 0.0)),
```

```
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Bend, Deschutes County, Oregon, United States, (44.0581728,
-121.3153096, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 None,
 None,
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Florida, United States, (27.7567667, -81.4639835, 0.0)),
 Location(Spokane, Spokane County, Washington, United States, (47.6571934,
-117.42351, 0.0)),
 Location(Chehalis, Lewis County, Washington, United States, (46.6599653,
-122.963432, 0.0)),
 Location(Post Falls, Kootenai County, Idaho, 83854, United States, (47.7110106,
-116.9480877, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(New York, United States, (40.7127281, -74.0060152, 0.0)),
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Portland, Multnomah County, Oregon, United States, (45.5202471,
-122.674194, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Washougal, Clark County, Washington, 98671, United States,
(45.5819594, -122.3479921, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
```

```
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Plymouth, England, United Kingdom, (50.3712659, -4.1425658, 0.0)),
 None,
 None,
 Location(Richland, Benton County, Washington, United States, (46.2804298,
-119.275205, 0.0)),
 None,
 Location(Macon County, Illinois, United States, (39.8455208, -88.9602376,
0.0)),
 None,
 None,
 None,
 Location(E.R.F., 38, Kaiser-Franz-Josef-Kai, Innere Stadt, Graz, Styria, 8010,
Austria, (47.0747655, 15.4350534, 0.0)),
 None,
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(New York, United States, (43.1561681, -75.8449946, 0.0)),
 Location(South Sound, Scotland, United Kingdom, (60.5882108, -1.0024167, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(São Paulo, Região Imediata de São Paulo, Região Metropolitana de São
Paulo, Região Geográfica Intermediária de São Paulo, São Paulo, Southeast
Region, Brazil, (-23.5506507, -46.6333824, 0.0)),
 None,
 None,
 Location(University at Albany, The State University of New York, 1400,
Washington Avenue, City of Albany, Albany County, New York, 12222, United
States, (42.6840271, -73.82587727551194, 0.0)),
 None,
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Cmentarz wojenny nr 206 - Pogórska Wola, 73;94, Pogórska Wola, gmina
```

```
Skrzyszów, Tarnów County, Lesser Poland Voivodeship, 33-152, Poland, (50.020722,
21.166861, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Redmond, King County, Washington, United States, (47.6694141,
-122.1238767, 0.0)),
 Location(Iowa City, Johnson County, Iowa, United States, (41.6612561,
-91.5299106, 0.0)),
 Location(Queensland, Australia, (-22.1646782, 144.5844903, 0.0)),
 None,
 None,
 Location(Florida, United States, (27.7567667, -81.4639835, 0.0)),
 Location(Washington, United States, (47.2868352, -120.212613, 0.0)),
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 Location(New York, United States, (40.7127281, -74.0060152, 0.0)),
 None,
 None,
 None,
 None,
 Location(Mill Creek, Snohomish County, Washington, United States, (47.8577455,
-122.2210628, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Kailua-Kona, Hawai i County, Hawaii, 96840, United States, (19.641941,
-155.996933, 0.0)),
 Location(Hutchinson, McLeod County, Minnesota, 55350, United States,
(44.8877401, -94.3697057, 0.0)),
 Location(Hartford, Capitol Region COG, Connecticut, United States, (41.764582,
-72.6908547, 0.0)),
 None,
 None,
 None,
 Location(Tacoma, Pierce County, Washington, United States, (47.2455013,
-122.438329, 0.0)),
 None,
 Location(West Seattle, Seattle, King County, Washington, 98116, United States,
(47.5709315, -122.3865166, 0.0)),
 None,
 Location(Bend, Deschutes County, Oregon, United States, (44.0581728,
-121.3153096, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
```

-122.330062, 0.0)),
 Location(London, Greater London, England, United Kingdom, (51.5073359,
-0.12765, 0.0)),
 Location(Iowa City, Johnson County, Iowa, United States, (41.6612561,
-91.5299106, 0.0)),
 Location(Middle Earth, University of California, Irvine, Irvine, Orange County,
California, United States, (33.644680199999996, -117.83728651980151, 0.0)),
 Location(Chicago, Cook County, Illinois, United States, (41.8755616,
-87.6244212, 0.0)),
 Location(Puyallup, Pierce County, Washington, United States, (47.1849009,
-122.29214, 0.0)),
 None,
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Los Angeles, Los Angeles County, California, United States,
(34.0536909, -118.242766, 0.0)),
 Location(Marysville, Snohomish County, Washington, United States, (48.0517429,
-122.1768209, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 Location(Las Vegas, Clark County, Nevada, United States, (36.1672559,
-115.148516, 0.0)),
 Location(Newcastle, Young County, United States, (33.192915, -98.7375358,
0.0)),
 None,
 None,
 Location(Oregon, Municipality of Victoria County, Victoria County, Nova Scotia,
Canada, (46.3187056, -60.6411688, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 Location(Fresno, Fresno County, California, United States, (36.7394421,
-119.7848307, 0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 None,
 None,
 None,
 Location(Iowa City, Johnson County, Iowa, United States, (41.6612561,
-91.5299106, 0.0)),
 Location(Whidbey Island, Island County, Washington, United States,
(48.15814255, -122.59742939440014, 0.0)),
 None,
 None,
 None,

```
 Location(The Emerald, Surfers Paradise, Gold Coast City, Queensland, Australia,
(-28.006135049999997, 153.42387460889537, 0.0)),
 Location(Washington, District of Columbia, United States, (38.8950368,
-77.0365427, 0.0)),
 None,
 Location(Zaandam, Zaanstad, North Holland, Netherlands, (52.4424926, 4.8298607,
0.0)),
 Location(Seattle, King County, Washington, United States, (47.6038321,
-122.330062, 0.0)),
 Location(Bellevue, King County, Washington, United States, (47.6144219,
-122.192337, 0.0)),
 None,
 None,
 None,
 Location(Newcastle, King County, Washington, United States, (47.5395736,
-122.1563331, 0.0)),
 …]
```

[21]:
```
lon = [x.longitude for x in map_location if x is not None ]
lat = [x.latitude for x in map_location if x is not None ]
temperature = [x.longitude for x in map_location if x is not None ]
```

[ ]:
```

```

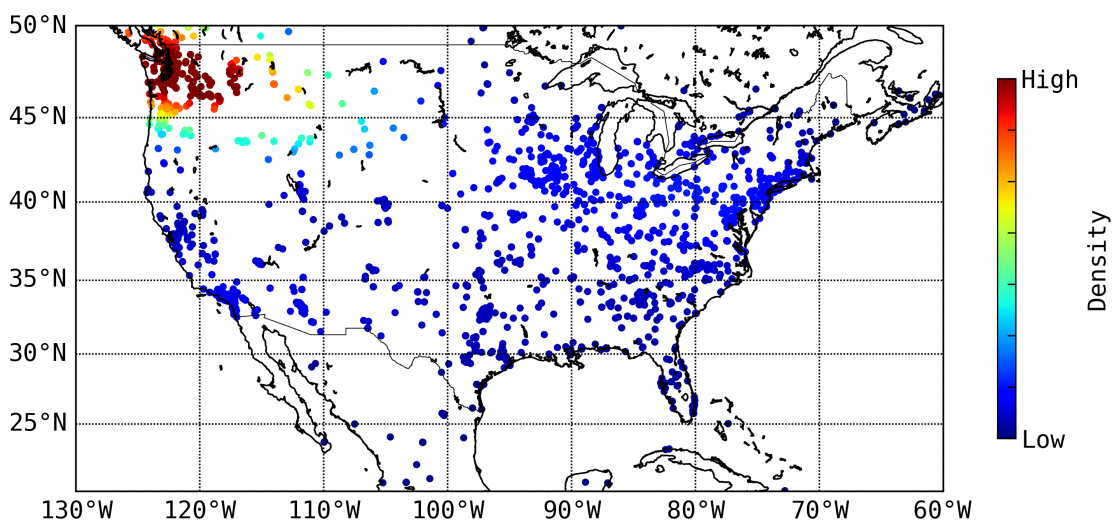
[22]:
```
from scipy.stats import gaussian_kde

# Calculate the point density
xy = np.vstack([lon,lat])
z = gaussian_kde(xy)(xy)
```

[36]:
```
lower, upper = 0, 40

width = upper-lower
minn = min(z)
maxx = max(z)

l_norm = [(x - minn)/(maxx - minn) * width + lower for x in z]
```

[37]:
```
%matplotlib inline
# import matplotlib as mpl
# import matplotlib.pyplot as plt
from mpl_toolkits.basemap import Basemap
# import numpy as np
from scipy.stats import gaussian_kde


m = Basemap(projection='merc',llcrnrlat=20,urcrnrlat=50,\
```

```
            llcrnrlon=-130,urcrnrlon=-60,lat_ts=20,resolution='i')
m.drawcoastlines()
m.drawcountries()
# draw parallels and meridians.
parallels = np.arange(-90.,91.,5.)
# Label the meridians and parallels
m.drawparallels(parallels,labels=[True,False,False,False])
# Draw Meridians and Labels
meridians = np.arange(-180.,181.,10.)
m.drawmeridians(meridians,labels=[True,False,False,True])
m.drawmapboundary(fill_color='white')
# plt.title("Forecast {0} days out".format(day_out))
# Define a colormap
jet = plt.cm.get_cmap('jet')
# Transform points into Map's projection
x,y = m(lon, lat)
# Color the transformed points!
sc = plt.scatter(x,y, c=l_norm, vmin=0, vmax =35, cmap=jet, s=20,␣
 ↪edgecolors='none')
# And let's include that colorbar
cbar = plt.colorbar(sc, shrink = .5)
cbar.ax.set_yticklabels(['Low', "", "", "", '', "", "", 'High'])  # vertically␣
 ↪oriented colorbar
cbar.set_label('Density')
plt.show()
```

```
/tmp/ipykernel_27643/2622752309.py:30: UserWarning: FixedFormatter should only
be used together with FixedLocator
  cbar.ax.set_yticklabels(['Low', "", "", "", '', "", "", 'High'])  # vertically
oriented colorbar
```



[25]:
```
len(lat)
```

[25]: 20950

[26]:
```
min(temperature)
```

[26]: -170.7131481

[27]:
```
max(temperature)
```

[27]: 178.1367523749267

[ ]:
```

```

[ ]:
```

```

P4_Q10_EDA_Patriors

March 9, 2023

[1]:
```
# !pip install latex
```

[5]:
```
import datetime
import json
import math
import os
import pytz
import random
import re

from functools import reduce, partial
from joblib import Memory
from tempfile import mkdtemp
from typing import List

import nltk
import numpy as np
import pandas as pd
import umap.umap_ as umap
import umap.plot

from matplotlib import pyplot as plt
from tqdm.auto import tqdm
```

[16]:
```
SEED = 806
np.random.seed(SEED)
# plt.style.use("seaborn-bright")
random.seed(SEED)

import matplotlib as mpl
mpl.rcParams.update(mpl.rcParamsDefault)


import matplotlib.pylab as pylab
params = {'legend.fontsize': 'x-large',
          'figure.figsize': (15, 5),
         'axes.labelsize': 'x-large',
```

```
         'axes.titlesize':'x-large',
         'xtick.labelsize':'x-large',
         'ytick.labelsize':'x-large'}
pylab.rcParams.update(params)

fsize = 15
tsize = 18
tdir = 'in'
major = 5.0
minor = 3.0
lwidth = 0.8
lhandle = 2.0
plt.style.use('default')
plt.rcParams['text.usetex'] = True
plt.rcParams['font.size'] = fsize
plt.rcParams['legend.fontsize'] = tsize
plt.rcParams['xtick.direction'] = tdir
plt.rcParams['ytick.direction'] = tdir
plt.rcParams['xtick.major.size'] = major
plt.rcParams['xtick.minor.size'] = minor
plt.rcParams['ytick.major.size'] = 5.0
plt.rcParams['ytick.minor.size'] = 3.0
plt.rcParams['axes.linewidth'] = lwidth
plt.rcParams['legend.handlelength'] = lhandle

colors = ['#0051a2', '#97964a', '#ffd44f', '#f4777f', '#93003a']


plt.rcParams[u'figure.figsize'] = [12, 8]
plt.rcParams[u'figure.dpi'] = 300
```

[ ]:

```

```

[7]:

```
game_start = 1422833400
game_end = 1422845700
```

[8]:

```
import pytz
pst_tz = pytz.timezone('America/Los_Angeles')
datetime_object_in_pst_timezone = datetime.datetime.fromtimestamp(game_start,␣
 ↪pst_tz)
datetime_object_in_pst_timezone
```

[8]:

```
datetime.datetime(2015, 2, 1, 15, 30, tzinfo=<DstTzInfo 'America/Los_Angeles'
PST-1 day, 16:00:00 STD>)
```

[9]:

```
import pytz
pst_tz = pytz.timezone('America/Los_Angeles')
```

```
datetime_object_in_pst_timezone = datetime.datetime.fromtimestamp(game_end,␣
↪pst_tz)
datetime_object_in_pst_timezone
```

[9]:
```
datetime.datetime(2015, 2, 1, 18, 55, tzinfo=<DstTzInfo 'America/Los_Angeles'
PST-1 day, 16:00:00 STD>)
```

[11]:
```
data_dir = "./ECE219_tweet_data"
filename = "tweets_#superbowl.txt"
data = []
n = 100

with open(os.path.join(data_dir, filename)) as f:
    for line in tqdm(f):
        all_info = json.loads(line)
        # print(all_info.keys())
        # print(all_info["tweet"]["user"])
        taken_info = {
            "type": all_info["type"],
            # "title": all_info["title"],
            "text": all_info["tweet"]["text"],
            "id": all_info["tweet"]["id"],
            "hashtags": " ".join([
                " ".join([s for s in re.split('([A-Z][a-z]+)', h["text"]) if s])
                for h in all_info["tweet"]["entities"]["hashtags"]
            ]),  # hashtag split
            "location": all_info["tweet"]["user"]['location'],
            "citation_date": all_info["citation_date"],
        }
        if taken_info["citation_date"] > game_end:
            continue
        if taken_info["citation_date"] <= game_start or taken_info["text"] ==␣
↪"":
            continue
        data.append(taken_info)

data_df = pd.DataFrame(data)
```

```
0it [00:00, ?it/s]
```

[ ]:
```

```

[12]:
```
data_df['citation_date_datetime'] = data_df['citation_date'].apply(datetime.
↪datetime.fromtimestamp, args=(pst_tz,))
```

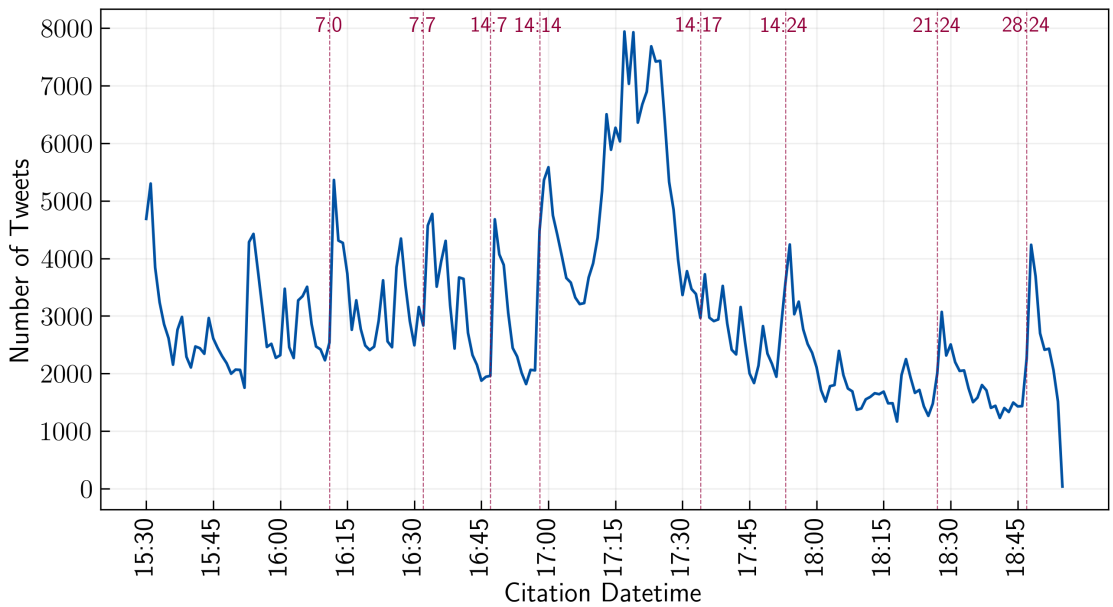
[13]:
```
def tweets_per_minute(df: pd.DataFrame):
    return df[['text']].groupby([df['citation_date_datetime'].dt.date,
```

```
                                    df['citation_date_datetime'].dt.hour,
                                    df['citation_date_datetime'].dt.minute,
                                   ]).count()
```

[106]:
```
import matplotlib.colors as mcolors

def tweets_per_minute_plot(df: pd.DataFrame, title_prefix="", sample_freq=15):

    group_df = tweets_per_minute(df)
    fig = plt.figure(figsize = (10, 5))

    cmap = mcolors.LinearSegmentedColormap.from_list("", ["gold", "gold",␣
 ↪"green", "darkblue"])

    plt.plot([f"{x[0]} {x[1]}H {x[2]}M" for x in group_df.index],
             group_df.melt()['value'],
#              width=1,
#              color=cmap(group_df.melt()['value']/group_df.melt()['value'].
 ↪max())
             color=colors[0]
            )
    max_value = (max(group_df.melt()['value']))

    scoring_timestamp = [1422835905, 1422837178, 1422838078, 1422838706,␣
 ↪1422840875, 1422842032, 1422844038, 1422845270]

    scoring_text = ["7:0", "7:7", "14:7",
                    "14:14", "14:17", "14:24", "21:24", "28:24"]
    for idx, timestamp in enumerate(scoring_timestamp):
        dt = datetime.datetime.fromtimestamp(timestamp, pst_tz)
        plt.axvline(list(group_df.index).index((dt.date(), dt.hour, dt.minute)),
                    color=colors[-1], linestyle='--', alpha=0.7, linewidth=0.6)
#         plt.text(list(group_df.index).index((dt.date(), dt.hour, dt.
 ↪minute))+5,
#                  max_value+1, str(idx+1),
#                  fontsize=10,
#                  ha="center",
#                  va="center",
#                  bbox = dict(boxstyle=f"circle,pad={0.2}", fc="white",␣
 ↪color="red"), color="red")
        plt.text(list(group_df.index).index((dt.date(), dt.hour, dt.minute)),
                 max_value,
                 scoring_text[idx],
                 fontfamily='monospace',
                 horizontalalignment='center',
                 fontsize=12,
                 color=colors[-1])
```

```
    plt.xticks(np.arange(0, len(group_df.index), step=sample_freq),
               [f"{x[1]}:{x[2]:02d}" for x in group_df.index][::sample_freq],
               rotation=90)
    plt.xlabel("Citation Datetime")
    plt.ylabel("Number of Tweets")
    plt.grid(alpha = .2)

    plt.show()
```

[107]:
```
tweets_per_minute_plot(data_df, title_prefix="Super Bowl")
```



[ ]:

[ ]:

[91]:
```
def read_file(filename: str):
    data_dir = ""
    data = []
    n = 100

    with open(os.path.join(data_dir, filename)) as f:
        for line in tqdm(f):
            all_info = json.loads(line)
            taken_info = {
```

```
                "type": all_info["type"],
                "title": all_info["title"],
                "text": all_info["tweet"]["text"],
                "id": all_info["tweet"]["id"],
                "hashtags": [
                    h["text"].lower() for h in␣
↪all_info["tweet"]["entities"]["hashtags"]
                ],
                "retweet_count": all_info["tweet"]["retweet_count"],
                "author_followers": all_info["author"]["followers"],
                "original_author_followers":␣
↪all_info["original_author"]["followers"],
                "citation_date": all_info["citation_date"],
                "ranking_score": all_info["metrics"]["ranking_score"],
                "citations": all_info["metrics"]["citations"]["total"],
                "impressions": all_info["metrics"]["impressions"],
            }
            data.append(taken_info)
    #         if len(data) == n:
    #             break

    data_df = pd.DataFrame(data)
    data_df['citation_date_datetime'] = data_df['citation_date'].apply(datetime.
↪datetime.fromtimestamp, args=(pst_tz,))
    return data_df

def get_filepaths(rootdir: str):
    res = []
    for subdir, dirs, files in os.walk(rootdir):
        for file in files:
            filepath = subdir + os.sep + file
            if filepath.endswith(".txt"):
                res.append(filepath)
    return res

def file_statistics(df: str):
    pst_tz = pytz.timezone('America/Los_Angeles')
    df['citation_date_datetime'] = df['citation_date'].apply(datetime.datetime.
↪fromtimestamp, args=(pst_tz,))
    max_citation_date = max(df['citation_date_datetime'])
    min_citation_date = min(df['citation_date_datetime'])

    avg_tweets_per_hour = len(df) / ((max_citation_date - min_citation_date).
↪total_seconds() / 60 / 60)
    avg_followers_posting_tweets = df['citations'].sum() / len(df)
    avg_retweet_per_tweet = df['author_followers'].sum() / len(df)
```

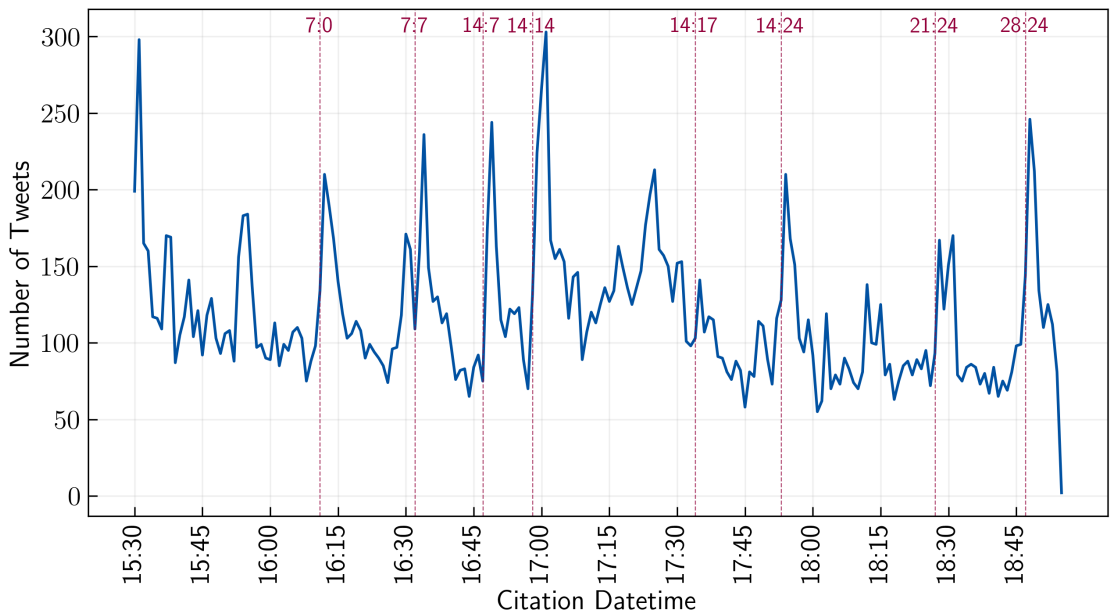
```
    print(f'Average tweets per minute: {avg_tweets_per_hour}')
    print(f'Average number of users posting tweets per tweet:␣
↪{avg_followers_posting_tweets}')
    print(f'Average number of retweet per tweet: {avg_retweet_per_tweet}')
```

[92]:
```
rootdir = "./ECE219_tweet_data"
dataset = {}

filepaths = get_filepaths(rootdir)
dataset_dict = {filepath: read_file(filepath) for filepath in filepaths}
```

```
0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]

0it [00:00, ?it/s]
```

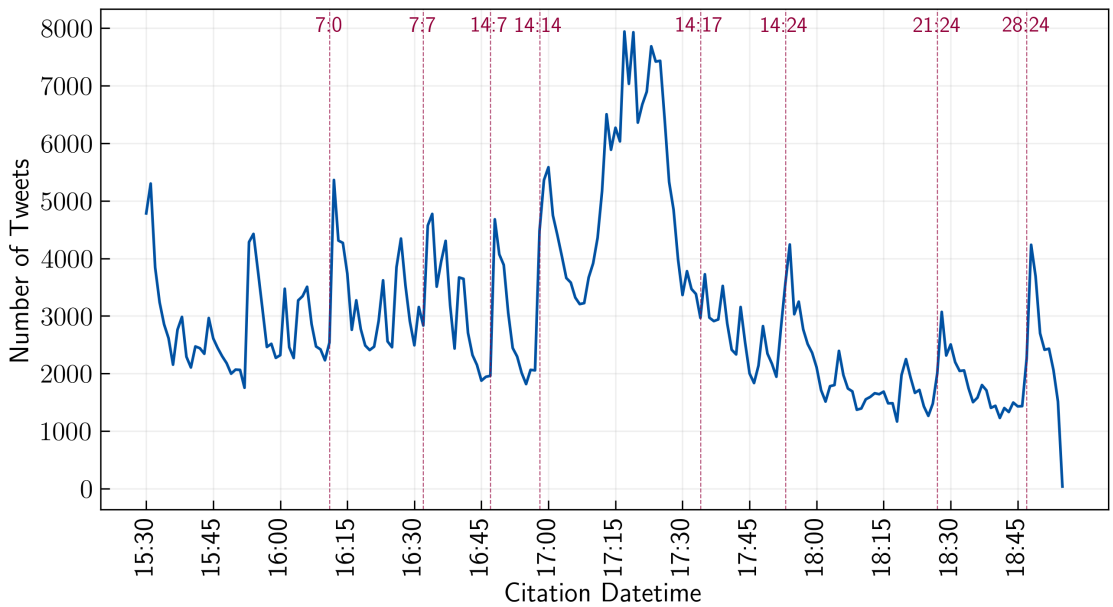
[93]:
```
for filepath in filepaths:
    print(filepath)
    df = dataset_dict[filepath]
    df = df[(df['citation_date'] >= game_start) & (df['citation_date'] <=␣
↪game_end)]
    df['citation_date_datetime'] = df['citation_date'].apply(datetime.datetime.
↪fromtimestamp, args=(pst_tz,))
    tweets_per_minute_plot(df,
                           title_prefix=filepath.split("#")[1][:-4])
```

```
./ECE219_tweet_data/tweets_#nfl.txt

/var/folders/tw/c9bx4g3s1gz_0c64pb22gklm0000gn/T/ipykernel_92199/2362341121.py:5
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```



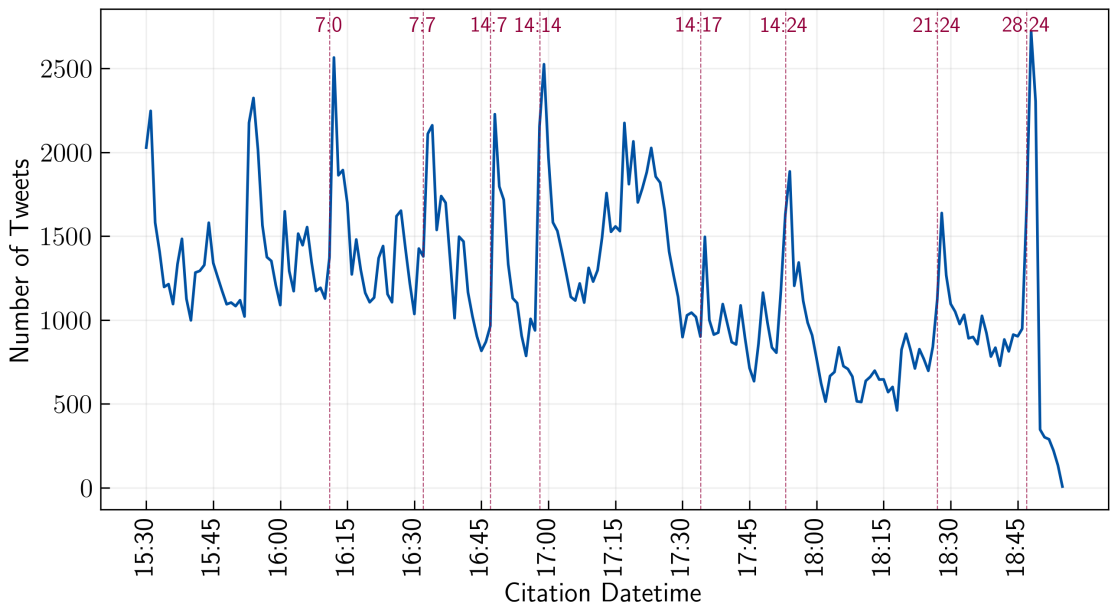

```
./ECE219_tweet_data/tweets_#superbowl.txt
```

```
/var/folders/tw/c9bx4g3s1gz_0c64pb22gklm0000gn/T/ipykernel_92199/2362341121.py:5
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```



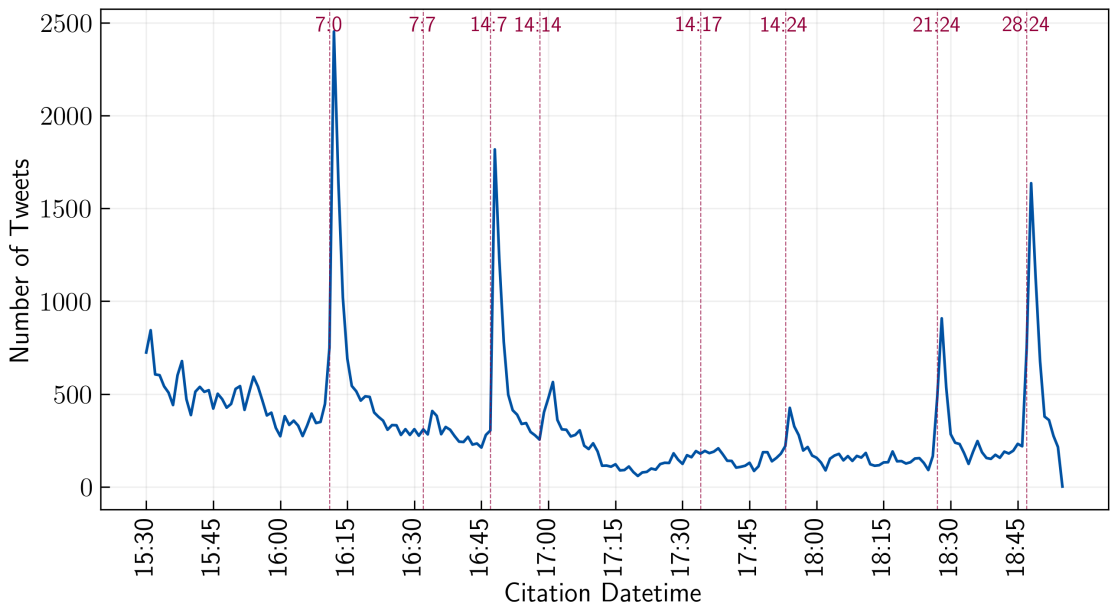

```
./ECE219_tweet_data/tweets_#sb49.txt
```

```
/var/folders/tw/c9bx4g3s1gz_0c64pb22gklm0000gn/T/ipykernel_92199/2362341121.py:5
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```



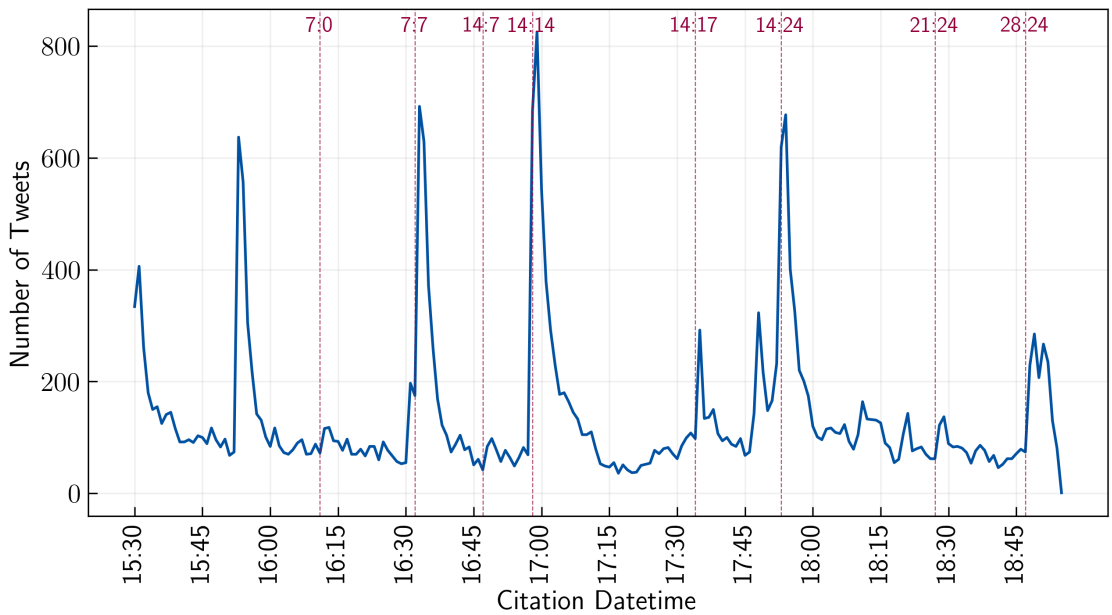

```
./ECE219_tweet_data/tweets_#patriots.txt
```

```
/var/folders/tw/c9bx4g3s1gz_0c64pb22gklm0000gn/T/ipykernel_92199/2362341121.py:5
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```



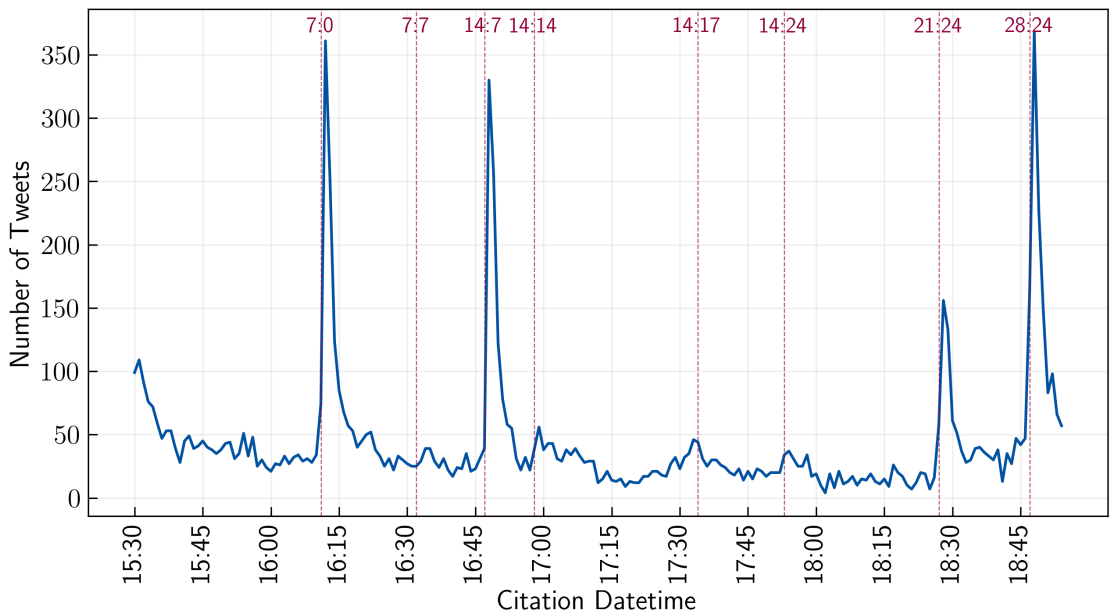

```
./ECE219_tweet_data/tweets_#gohawks.txt
```

```
/var/folders/tw/c9bx4g3s1gz_0c64pb22gklm0000gn/T/ipykernel_92199/2362341121.py:5
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```




```
./ECE219_tweet_data/tweets_#gopatriots.txt
```

```
/var/folders/tw/c9bx4g3s1gz_0c64pb22gklm0000gn/T/ipykernel_92199/2362341121.py:5
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['citation_date_datetime'] =
df['citation_date'].apply(datetime.datetime.fromtimestamp, args=(pst_tz,))
```



[ ]:

```
```

[ ]:

```
```

[15]:

```
data_dir = "/tmp2/kch/ece219/ECE219_tweet_data"
filename = "tweets_#gopatriots.txt"
data = []
n = 100

with open(os.path.join(data_dir, filename)) as f:
    for line in tqdm(f):
        all_info = json.loads(line)
        # print(all_info["tweet"]["user"])
        taken_info = {
#             "type": all_info["type"],
#             # "title": all_info["title"],
            "text": all_info["tweet"]["text"],
            "id": all_info["tweet"]["id"],
#             "hashtags": " ".join([
#                 " ".join([s for s in re.split('([A-Z][a-z]+)', h["text"]) if␣
 ↪s])
#                 for h in all_info["tweet"]["entities"]["hashtags"]
#             ]),  # hashtag split
            "location": all_info["tweet"]["user"]['location'],
#             "citation_date": all_info["citation_date"],
        }
```

```
        data.append(taken_info)

data_df = pd.DataFrame(data)
```

```
0it [00:00, ?it/s]
```

[16]:
```
data_df
```

[16]:
```
                                                   text                  id  \
0      LeGarrette Blount does the Ray Lewis Dance #Th…  553650101105606656
1      Y así será mi futuro hijo     #GoPatriots #M…  553994483457875968
2      With both my father and husband yelling at the…  554076243038326785
3      The girls soccer team is ranked the No. 3 team…  554774415914061825
4      @patriots Patriots Champions! RT if you agree …  554801542201884672
…                                                   …                   …
23506  Butler!!!! Yeess!!! Esto si que sería ganar so…  563078525255958528
23507  Butler!!!! Yeess!!! Esto si que sería ganar so…  563137745967542274
23508  #PatriotsNation sign petition to help sad peop…  563453742960242689
23509  RT JomaGh: Comienza el #SUperBowl ¿A quien le …  563688095552663554
23510            That's how you do it. #sb49 #gopatriots  563969055279677440

                        location
0          Boston, Massachusetts
1
2                 new york city.
3                 Plantation, FL
4                    Dol Guldur.
…                              …
23506
23507                 Cuernavaca
23508  Seattle, WA  :  Tokyo, JP
23509             Cancun, Mexico
23510

[23511 rows x 3 columns]
```

[17]:
```
from geopy.exc import GeocoderTimedOut
from geopy.geocoders import Nominatim

from tqdm.auto import tqdm

map_location = []

# Used to map the twitter user location description to a standard format
def lookup_location(location, attempt=1, max_attempts=5):
    geo_locator = Nominatim(user_agent="LearnPython", timeout=None)
    try:
```

```
        location = geo_locator.geocode(location, language='en')
    except GeocoderTimedOut:
        if attempt <= max_attempts:
            return lookup_location(location, attempt=attempt+1)
        raise
    except:
        return
    return location
```

[18]:
```
# data_df['map_location'] = data_df['location'].apply(lookup_location)
```

[ ]:
```

```

[19]:
```
for loc in tqdm(data_df['location']):
    # print(loc)
    map_location.append(lookup_location(loc))
```

```
  0%|          | 0/23511 [00:00<?, ?it/s]
```

[20]:
```
# map_location
import pickle

# Its important to use binary mode
dbfile = open('map_location.pkl', 'ab')

# source, destination
pickle.dump(map_location, dbfile)
dbfile.close()
```

[21]:
```
dbfile = open('map_location.pkl', 'rb')
db = pickle.load(dbfile)
```

[22]:
```
# db
```

[23]:
```
lon = [x.longitude for x in map_location if x is not None ]
lat = [x.latitude for x in map_location if x is not None ]
temperature = [x.longitude for x in map_location if x is not None ]
```

[24]:
```
lower, upper = 0, 115

width = upper-lower
minn = min(temperature)
maxx = max(temperature)

l_norm = [(x - minn)/(maxx - minn) * width + lower for x in temperature]
```

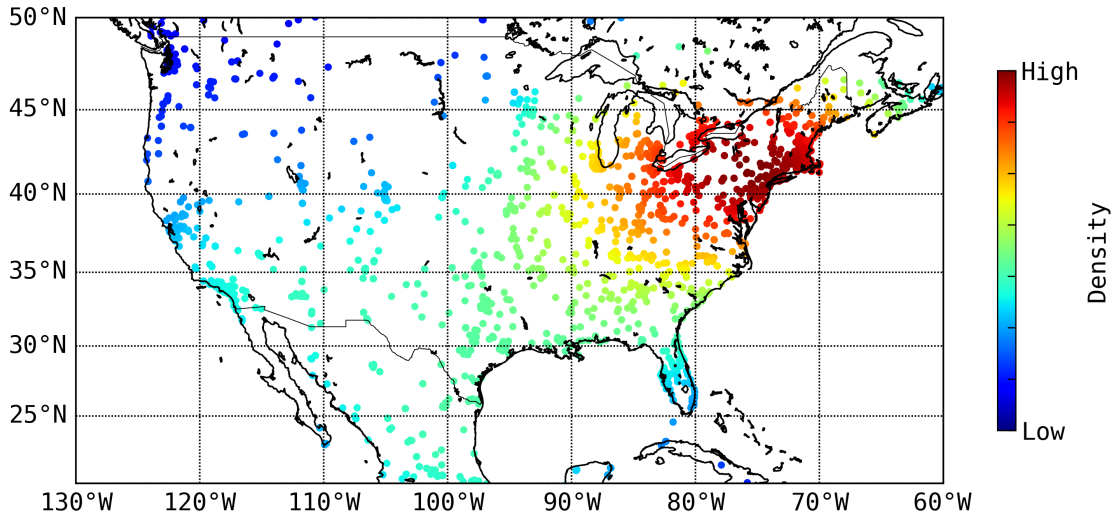
[25]:
```
from scipy.stats import gaussian_kde

# Calculate the point density
xy = np.vstack([lon,lat])
z = gaussian_kde(xy)(xy)
```

[26]:
```
lower, upper = 0, 35

width = upper-lower
minn = min(z)
maxx = max(z)

l_norm = [(x - minn)/(maxx - minn) * width + lower for x in z]
```

[54]:
```
%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
from mpl_toolkits.basemap import Basemap
import numpy as np
from scipy.stats import gaussian_kde


m = Basemap(projection='merc',llcrnrlat=20,urcrnrlat=50,\
            llcrnrlon=-130,urcrnrlon=-60,lat_ts=20,resolution='i')
m.drawcoastlines()
m.drawcountries()
# draw parallels and meridians.
parallels = np.arange(-90.,91.,5.)
# Label the meridians and parallels
m.drawparallels(parallels,labels=[True,False,False,False])
# Draw Meridians and Labels
meridians = np.arange(-180.,181.,10.)
m.drawmeridians(meridians,labels=[True,False,False,True])
m.drawmapboundary(fill_color='white')
# plt.title("Forecast {0} days out".format(day_out))
# Define a colormap
jet = plt.cm.get_cmap('jet')
# Transform points into Map's projection
x,y = m(lon, lat)
# Color the transformed points!
sc = plt.scatter(x,y, c=l_norm, vmin=0, vmax =35, cmap=jet, s=20,␣
 ↪edgecolors='none')
# And let's include that colorbar
cbar = plt.colorbar(sc, shrink = .5)
cbar.ax.set_yticklabels(['Low', "", "", "", '', "", "", 'High'])  # vertically␣
 ↪oriented colorbar
cbar.set_label('Density')
```

```
plt.show()
```

```
/tmp/ipykernel_27727/2957301837.py:30: UserWarning: FixedFormatter should only
be used together with FixedLocator
  cbar.ax.set_yticklabels(['Low', "", "", "", '', "", "", 'High'])  # vertically
oriented colorbar
```



[ ]:

```

```

P4_Q10_Classification

March 9, 2023

[7]:
```
import datetime
import json
import math
import os
import pytz
import random
import re

from functools import reduce, partial
from joblib import Memory
from multiprocessing import Pool
from tempfile import mkdtemp
from typing import List

import nltk
import numpy as np
import pandas as pd
import umap.umap_ as umap
import umap.plot

import emoji
from matplotlib import pyplot as plt
from nltk import pos_tag
from nltk.corpus import wordnet
from nltk.stem import PorterStemmer, WordNetLemmatizer
from nltk.tokenize import word_tokenize
from sentence_transformers import SentenceTransformer
from sklearn import preprocessing, metrics
from sklearn.base import TransformerMixin, BaseEstimator
from sklearn.compose import ColumnTransformer
from sklearn.decomposition import NMF, TruncatedSVD
from sklearn.feature_extraction.text import CountVectorizer, TfidfTransformer
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import Pipeline
from sklearn.metrics import (accuracy_score,
                             auc,
                             confusion_matrix,
```

```
                              ConfusionMatrixDisplay,
                              f1_score,
                              precision_score,
                              RocCurveDisplay,
                              recall_score,
                              roc_curve,
                             )
from sklearn.model_selection import GridSearchCV, train_test_split
from sklearn.multiclass import OneVsRestClassifier, OneVsOneClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import LinearSVC
from sklearn.utils.extmath import randomized_svd
from textblob import TextBlob
from tqdm.auto import tqdm

os.environ["CUDA_VISIBLE_DEVICES"] = "0"
```

[8]:
```
nltk.download("punkt")
nltk.download('averaged_perceptron_tagger')
nltk.download('wordnet')
nltk.download('omw-1.4')

SEED = 806
np.random.seed(SEED)
plt.style.use("seaborn-bright")
random.seed(SEED)
```

```
[nltk_data] Downloading package punkt to /home/ray080186/nltk_data…
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /home/ray080186/nltk_data…
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
[nltk_data] Downloading package wordnet to
[nltk_data]     /home/ray080186/nltk_data…
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package omw-1.4 to
[nltk_data]     /home/ray080186/nltk_data…
[nltk_data]   Package omw-1.4 is already up-to-date!
/tmp/ipykernel_17259/3026732915.py:8: MatplotlibDeprecationWarning: The seaborn
styles shipped by Matplotlib are deprecated since 3.6, as they no longer
correspond to the styles shipped by seaborn. However, they will remain available
as 'seaborn-v0_8-<style>'. Alternatively, directly use the seaborn API instead.
  plt.style.use("seaborn-bright")
```

[9]:
```
game_start = 1422835961
game_end = 1422845700
```

[10]:
```
data_dir = "/tmp2/kch/ece219/ECE219_tweet_data"
filename = "tweets_#superbowl.txt"
data = []

with open(os.path.join(data_dir, filename)) as f:
    for line in tqdm(f):
        all_info = json.loads(line)
        # print(all_info.keys())
        # print(all_info["tweet"]["user"])
        taken_info = {
            "type": all_info["type"],
            # "title": all_info["title"],
            "text": all_info["tweet"]["text"],
            "id": all_info["tweet"]["id"],
            "hashtags": " ".join([
                " ".join([s for s in re.split('([A-Z][a-z]+)', h["text"]) if s])
                for h in all_info["tweet"]["entities"]["hashtags"]
            ]),  # hashtag split
            "location": all_info["tweet"]["user"]['location'],
            "citation_date": all_info["citation_date"],
        }
        if (
            game_start < taken_info["citation_date"] <= game_end
            and taken_info["text"] != ""
        ):
            data.append(taken_info)

data_df = pd.DataFrame(data)
```

```
0it [00:00, ?it/s]
```

[11]:
```
len(data_df)
```

[11]: 501976

[12]:
```
# "NE"
# "TIE"
# "SEA"

# leading_label is who is leading the game starting from the timestamp

# not sure if it is accurate since I am not able to
# transfer the time into absolute time using the following link
# https://www.espn.com/nfl/playbyplay/_/gameId/400749027
# I am wasting too much time here so I just use the rough time by searching and␣
↪checking from the tweet
```

```
scoring_timestamp = [
    game_start,
    1422837233,
    1422838078,
    1422838706,
    1422841071,
    1422842032,
    1422844038,
    1422845360,
]
leading_results = ['NE', 'TIE', 'NE', 'TIE', 'SEA', 'SEA', 'SEA', 'NE']
map_row_to_class = {"NE": 0, "SEA": 1, "TIE": 2}
conditions = []

for i in range(1, len(scoring_timestamp)):
    conditions.append(
        data_df["citation_date"].between(scoring_timestamp[i-1],␣
 ↪scoring_timestamp[i]-1)
    )
conditions.append(data_df["citation_date"] >= scoring_timestamp[-1])


data_df['leading_label'] = np.select(
    conditions,
    leading_results,
    np.nan
)
data_df['leading_label_cat'] = data_df.leading_label.map(map_row_to_class)

data_df["full_text"] = data_df["text"] + " " + data_df["hashtags"]
```

[13]:
```
data_df[["leading_label", "leading_label_cat"]].value_counts()
```

[13]:
```
leading_label  leading_label_cat
TIE            2                    241432
SEA            1                    149033
NE             0                    111511
dtype: int64
```

[14]:
```
# Replace special chars with space
__CHARS_TO_REPLACE = [",", ";", "\t", "...", "\"", "\n", "\xa0"]
__CHARS_TO_REPLACE_REGEX = re.compile("|".join(map(re.escape,␣
 ↪__CHARS_TO_REPLACE)))

# Adding heading ahd trailing space on special chars
__CHARS_ADD_SPACE = ["?", "-", ":", "_", "(", ")", "!", "/", ".", " ", " ", "@",␣
 ↪"©", "®", "℗"]
```

```
__CHARS_ADD_SPACE_REGEX = re.compile("|".join(map(re.escape,␣
↪__CHARS_ADD_SPACE)))

# Replace special pattern with space
__PATTERN_TO_REPLACE = [
    r'(\s(and)*\s*[0-9]*\s*(more|other)\s*items*\s)',  # eliminate other items
    r'(\s\(\s*\)\s)'  # remove empty brackets
]
__PATTERN_TO_REPLACE_REGEX = re.compile("|".join(__PATTERN_TO_REPLACE))

# Remove excess space
__REMOVE_EXCESS_SPACE_REGEX = re.compile(r'\s+')

# XML standard defines a valid character as
# Char ::= #x9 | #xA | #xD | [#x20-#xD7FF] | [#xE000-#xFFFD] |␣
↪[#x10000-#x10FFFF]
__RGX_XML_CNTRL_CHR = re.
↪compile(u'[^\u0020-\uD7FF\u0009\u000A\u000D\uE000-\uFFFD\U00010000-\U0010FFFF]+')

# Regex to match money value
__RGX_MONEY = re.compile(r'(?:[^\d]|\b)(?:\d{1,2},)?\d{1,4}(?:[.,]\d{1,2})\s*(?:
↪us\$|usd|\$)*(?:[^\d]|\b)|'
                         '(free$)|'
                         r'(?:us\$|\$|usd|us \$) *-? *\d{1,4}(?:[.,]\d{1,2})?
↪(\s*)(?:us\$|usd|\$)*(?:[^\d]|\b)|'
                         r'\d{1,4}(?:[.,]\d{1,2})? *(?:us\$|usd|\$)')
__REPLACE_RGX_MONEY = ' $#money#$ '

# Time related regex
__YEAR_NUM = r'(19[7-9][0-9]|20[0-2][0-9]|[0-2][0-9])'
__MONTH_SHORT = r'(jan|feb|mar|apr|may|jun|jul|aug|sep|oct|nov|dec)'
__MONTH_LONG =␣
↪r'(january|february|march|april|may|june|july|august|september|october|november|december)'
__MONTH = r'(?:%s)' % '|'.join([__MONTH_LONG, __MONTH_SHORT,␣
↪r'0[1-9]|[1-9]|1[0-2]'])
__DAY_NUM = r'([012][1-9]|[123][0-1]|[1-9])'
__DATE_SEP = r"(/|-|,|\s)"

# Add ? quantifier to solves "feb 3 2018" substitute to "$#date#$ $#number#$"␣
↪instead of "$#date#$" problem
__DATE_LIST = [
    f'({__MONTH}({__DATE_SEP}+){__DAY_NUM}(?:({__DATE_SEP}+){__YEAR_NUM})?)',
    f'({__DAY_NUM}({__DATE_SEP}+){__MONTH}(?:({__DATE_SEP}+){__YEAR_NUM})?)',
    f'({__MONTH}({__DATE_SEP}+){__YEAR_NUM}(?:({__DATE_SEP}+){__DAY_NUM})?)',
    f'({__YEAR_NUM}({__DATE_SEP}+){__MONTH}(?:({__DATE_SEP}+){__DAY_NUM})?)',
]
```

```
__RGX_MONTH = ' (?:%s) ' % '|'.join([__MONTH_LONG, __MONTH_SHORT])
__REPLACE_RGX_MONTH = ' $#month#$ '
__RGX_DATE = re.compile(r'\s(?:%s)\s' % '|'.join(__DATE_LIST))
__REPLACE_RGX_DATE = ' $#date#$ '
__RGX_TIME = re.compile(r'\s((0[1-9]|[1-9]|1[0-9]|2[0-4])(\s*\:
↪\s*([0-9]|[0-5][0-9])){1,2}\s*(?:am|pm|))\s')
__REPLACE_RGX_TIME = ' $#time#$ '

# DIGIT and Numbers
__RGX_DIGIT = re.compile(r'\d+')
__REPLACE_RGX_DIGIT = ' $#number#$ '
__RGX_DIGIT_TEXT = re.compile(
  ␣
↪r'(\d+[a-z]+\d+|[a-z]+\d+[a-z]+|[a-z]+\d+|\d+[a-z]+)*(\d+[a-z]+\d+|[a-z]+\d+[a-z]+|[a-z]+\d
__REPLACE_RGX_DIGIT_TEXT = ' $#textnum#$ '


def normalize_text(desc: str) -> str:
    """
    normalize the incoming description string for cme
    :param desc: description string
    :return: normalized description string
    """
    normalize_steps = [
        lambda s: " " + s.lower() + " ",
        lambda s: re.sub(__CHARS_TO_REPLACE_REGEX, " ", s),
        lambda s: re.sub(__CHARS_ADD_SPACE_REGEX, lambda m: " " + m.group(0) +␣
↪" ", s),
        lambda s: re.sub(__PATTERN_TO_REPLACE_REGEX, " ", s),
        lambda s: re.sub(__REMOVE_EXCESS_SPACE_REGEX, " ", s),
        lambda s: s.strip(),
        # lambda s: s if s else "$#empty#$"
        lambda s: s if s else " "
    ]
    return compose(*normalize_steps)(desc)


def normalize_text_advanced(text: str) -> str:
    """
    normalize the text string for suprabbx
    :param text: subject + sender string
    :return: normalized subject + sender string
    """
    normalize_steps = [
        lambda s: " ".join([a for a in re.split('([A-Z][a-z]+)', s) if a]),
        lambda s: emoji.demojize(s, language='en'),
```

```
        lambda s: " " + str(s).lower() + " ",
        lambda s: re.sub(__RGX_XML_CNTRL_CHR, " ", s),
        lambda s: re.sub(__RGX_MONEY, __REPLACE_RGX_MONEY, s),
        lambda s: re.sub(__CHARS_TO_REPLACE_REGEX, " ", s),
        lambda s: re.sub(__CHARS_ADD_SPACE_REGEX, lambda m: " " + m.group(0) +␣
↪" ", s),
        lambda s: re.sub(__PATTERN_TO_REPLACE_REGEX, " ", s),
        # Replace single space with multiple spaces to solve `xxx oct nov xxx`␣
↪problem
        lambda s: re.sub(" ", "  ", s),
        lambda s: re.sub(__RGX_TIME, __REPLACE_RGX_TIME, s),
        lambda s: re.sub(__RGX_DATE, __REPLACE_RGX_DATE, s),
        lambda s: re.sub(__RGX_MONTH, __REPLACE_RGX_MONTH, s),
        lambda s: re.sub(__RGX_DIGIT_TEXT, __REPLACE_RGX_DIGIT_TEXT, s),
        lambda s: re.sub(__RGX_DIGIT, __REPLACE_RGX_DIGIT, s),
        lambda s: re.sub(__REMOVE_EXCESS_SPACE_REGEX, " ", s),
        lambda s: s.strip(),
        # lambda s: s if s else "$#empty#$"
        lambda s: s if s else " "

    ]
    return compose(*normalize_steps)(text)


def compose(*functions):
    """
    compose multiple single argument functions together
    e.g., f3(f2(f1(x)))
    (f1 takes effect first, then f2, then f3)
    can be written as:
    func_list = [f1, f2, f3]
    compose(*func_list)(x)
    """
    return lambda x: reduce(lambda acc, f: f(acc), functions, x)
```

[15]:
```
def average(lst):
    if len(lst):
        return sum(lst) / len(lst)
    else:
        return 0

def get_sentiment(sen):
    sen_score = average(
        [
            s.sentiment.polarity
            for s in TextBlob(sen).sentences if s.sentiment.polarity != 0
        ]
```

```
    )

    if sen_score >= 0.5:
        return 3
    if sen_score > 0.0:
        return 1
    if sen_score == 0.0:
        return 0
    if sen_score > -0.5:
        return -1
    else:
        return -3
```

[16]:
```
def preprocess_text(text, stemming=True, normalize=True):
    if stemming:
        text = stem(text)
    if normalize:
        text = normalize_text_advanced(text)
    return text


def stem(text):
    stemmer = PorterStemmer()
    # Reference: https://www.geeksforgeeks.org/python-stemming-words-with-nltk/
    tokens = word_tokenize(text)
    text = " ".join([stemmer.stem(word.lower()) for word in tokens])
    return text
```

[17]:
```
def evaluate_mul(y_test, y_pred, average=None, clf=None):
    print("=" * 20)

    # confusion matrix
    labels = clf.classes_ if clf is not None else None
    cm = confusion_matrix(y_test, y_pred, labels=labels)
    cm_display = ConfusionMatrixDisplay(confusion_matrix=cm,␣
 ↪display_labels=labels)
    cm_display.plot()
    plt.show()

    acc = round(accuracy_score(y_test,y_pred), 5)

    print(f"Accuracy: {acc}")
    print(f"Recall: {recall_score(y_test, y_pred, average=None)}")
    print(f"Precision: {precision_score(y_test, y_pred, average=None)}")
    print(f"F-1 score: {f1_score(y_test, y_pred, average=None)}")
    print("=" * 20)
    print()
```

```
    return acc
```

[18]:
```
def get_train_test_indices(n, test_size):
    n_train = round(n * (1 - test_size))
    all_idxs = list(range(n))
    random.shuffle(all_idxs)
    return all_idxs[:n_train], all_idxs[n_train:]


train_idxs, test_idxs = get_train_test_indices(len(data_df), test_size=0.9)
train_mul, test_mul = data_df.iloc[train_idxs], data_df.iloc[test_idxs]
```

[13]:
```
def run_pipeline_and_get_accuracy(
    stemming, normalize, add_location, add_sentiment, dimension_reduction
):
    # preprocess text
    with Pool(24) as pool:
        train_mul["full_text_processed"] = pool.starmap(
            preprocess_text,
            [(t, stemming, normalize) for t in train_mul["full_text"]],
        )
        test_mul["full_text_processed"] = pool.starmap(
            preprocess_text,
            [(t, stemming, normalize) for t in test_mul["full_text"]],
        )

    # location
    if add_location:
        train_mul["full_text_processed"] += " " + train_mul["location"]
        test_mul["full_text_processed"] += " " + test_mul["location"]

    # tf-idf
    pipe = Pipeline(
        [
            ("count", CountVectorizer(stop_words="english", min_df=3)),
            ("tfidf", TfidfTransformer()),
        ]
    )
    X_train_mul = pipe.fit_transform(train_mul["full_text_processed"])
    X_test_mul = pipe.transform(test_mul["full_text_processed"])

    # dimension reduction
    n_components = 80
    if dimension_reduction == "nmf":
        model = NMF(init="random", n_components=n_components, random_state=SEED)
    elif dimension_reduction == "svd":
```

```
        model = TruncatedSVD(n_components=n_components, random_state=SEED)

    X_train_mul_reduced = model.fit_transform(X_train_mul)
    X_test_mul_reduced = model.transform(X_test_mul)

    # add sentiment feature
    if add_sentiment:
        with Pool(24) as pool:
            sentiments_train = pool.map(get_sentiment,␣
↪train_mul["full_text_processed"])
            sentiments_test = pool.map(get_sentiment,␣
↪test_mul["full_text_processed"])

        X_train_mul_reduced = np.concatenate(
            (X_train_mul_reduced, np.expand_dims(np.array(sentiments_train),␣
↪1)), axis=1
        )
        X_test_mul_reduced = np.concatenate(
            (X_test_mul_reduced, np.expand_dims(np.array(sentiments_test), 1)),␣
↪axis=1
        )

    # classifier
    one_one_svm = OneVsOneClassifier(LinearSVC(random_state=SEED))
    y_pred = one_one_svm.fit(
        X_train_mul_reduced, train_mul["leading_label_cat"]
    ).predict(X_test_mul_reduced)
    acc = evaluate_mul(
        test_mul["leading_label_cat"], y_pred, average=None, clf=one_one_svm
    )

    return acc
```

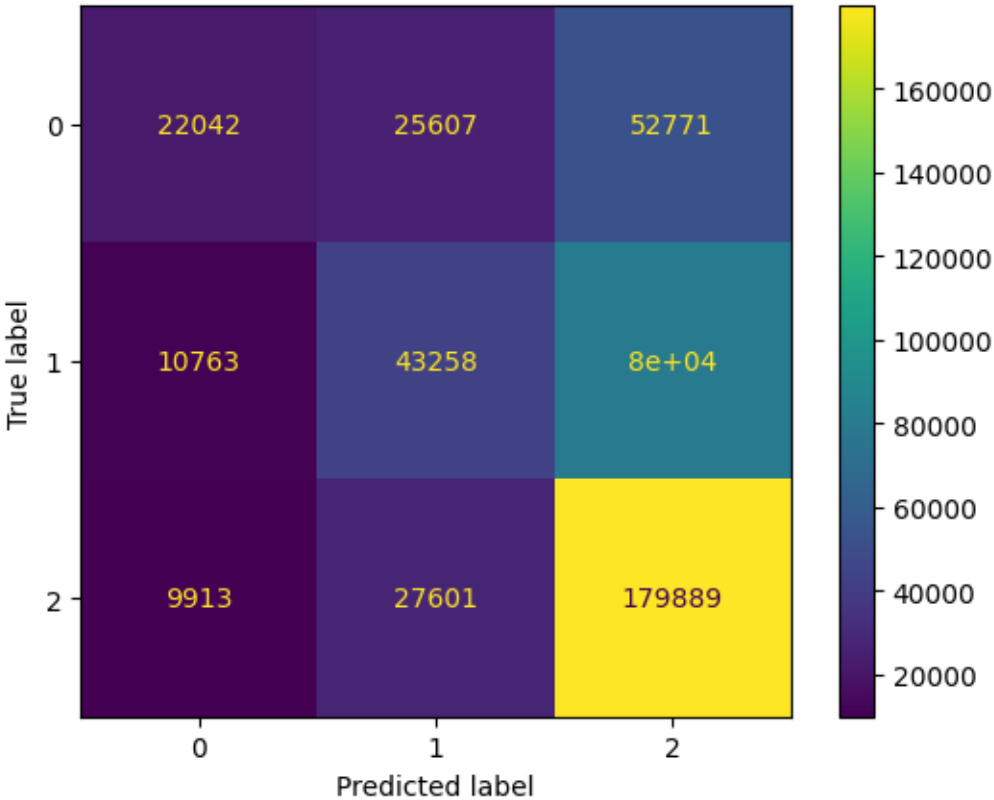
[14]:
```
params2acc = {}

for dimension_reduction in ["svd", "nmf"]:
    for normalize in [True, False]:
        for stemming in [True, False]:
            for add_sentiment in [True, False]:
                for add_location in [True, False]:
                    print(
                        f"{stemming=}, {normalize=}, {add_sentiment=}, "
                        f"{add_location=}, {dimension_reduction=}"
                    )
                    acc = run_pipeline_and_get_accuracy(
                        stemming,
                        normalize,
```

```
                        add_location,
                        add_sentiment,
                        dimension_reduction,
                    )
                    params2acc[
                        (
                            stemming,
                            normalize,
                            add_location,
                            add_sentiment,
                            dimension_reduction,
                        )
                    ] = acc

params2acc
```

```
stemming=True, normalize=True, add_sentiment=True, add_location=True,
dimension_reduction='svd'
```

```
/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]
```

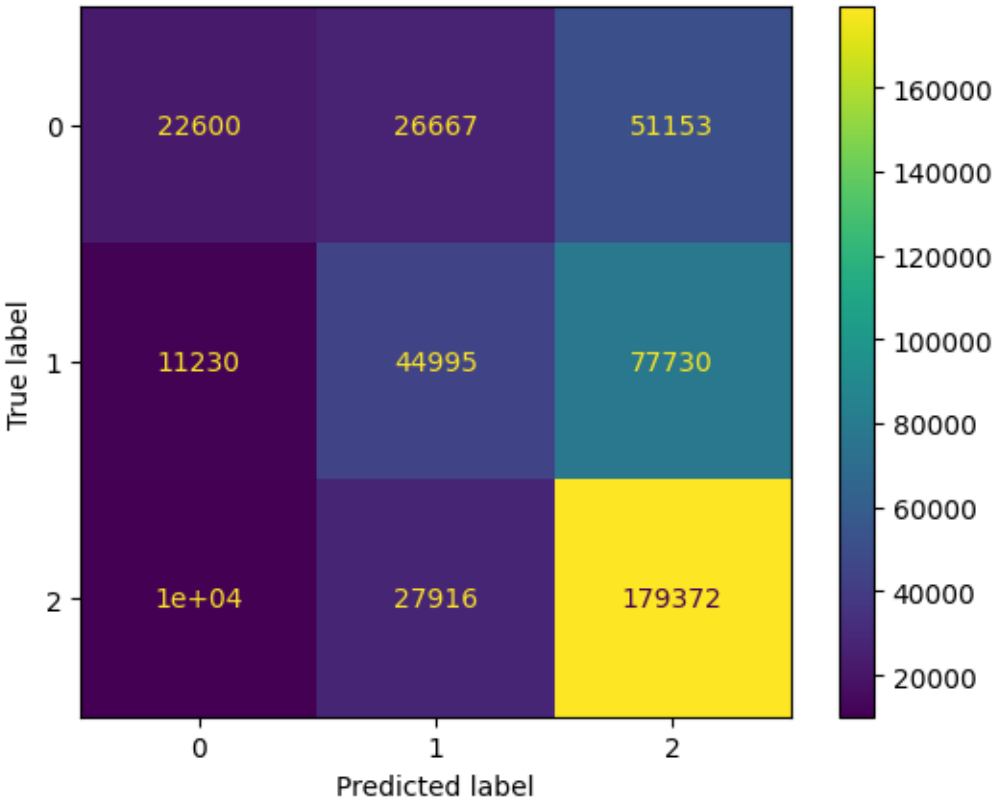
```
====================
```




```
Accuracy: 0.54272
Recall: [0.21949811 0.32292934 0.82744488]
Precision: [0.51598858 0.44842743 0.5754717 ]
F-1 score: [0.30798251 0.37546925 0.67883026]
====================

stemming=True, normalize=True, add_sentiment=True, add_location=False,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
```

==========

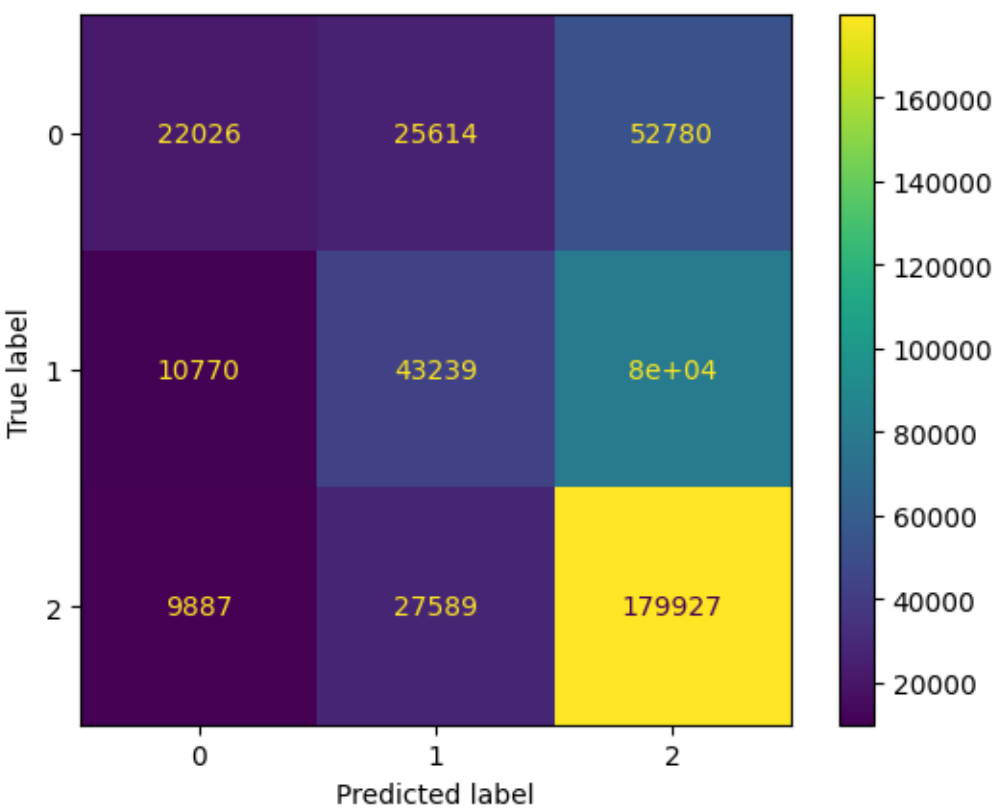



```
Accuracy: 0.54666
Recall: [0.22505477 0.33589638 0.82506681]
Precision: [0.51427921 0.45185684 0.58189486]
F-1 score: [0.31309528 0.38534169 0.68246655]
```

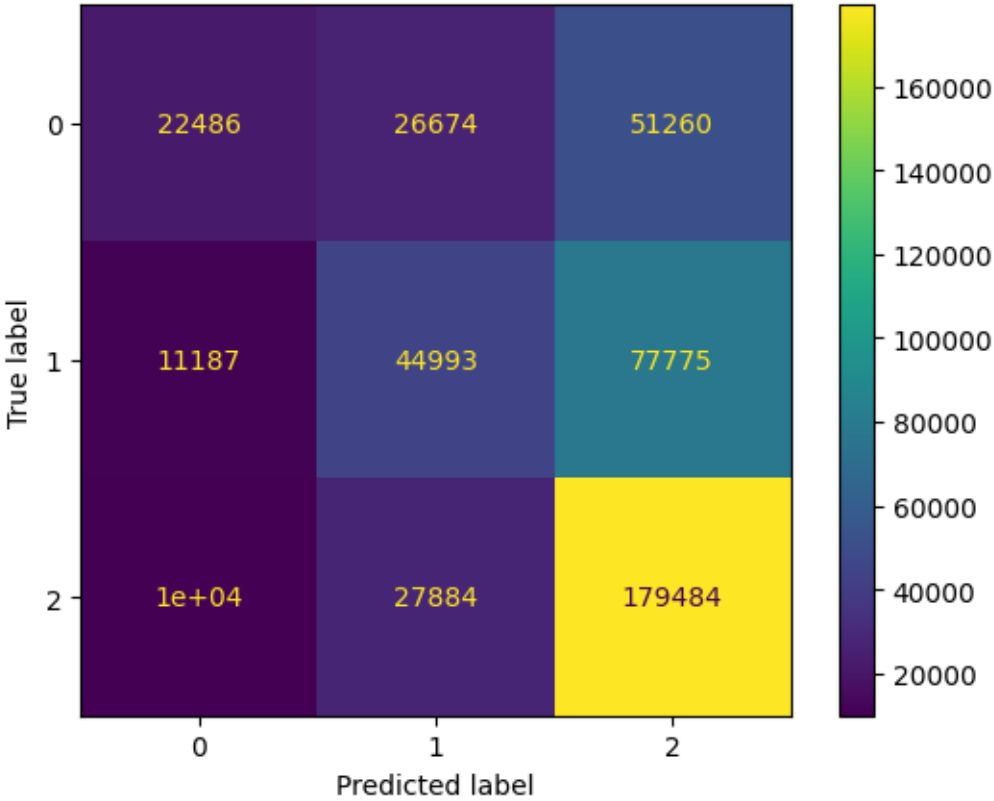
==========

```
stemming=True, normalize=True, add_sentiment=False, add_location=True,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
```

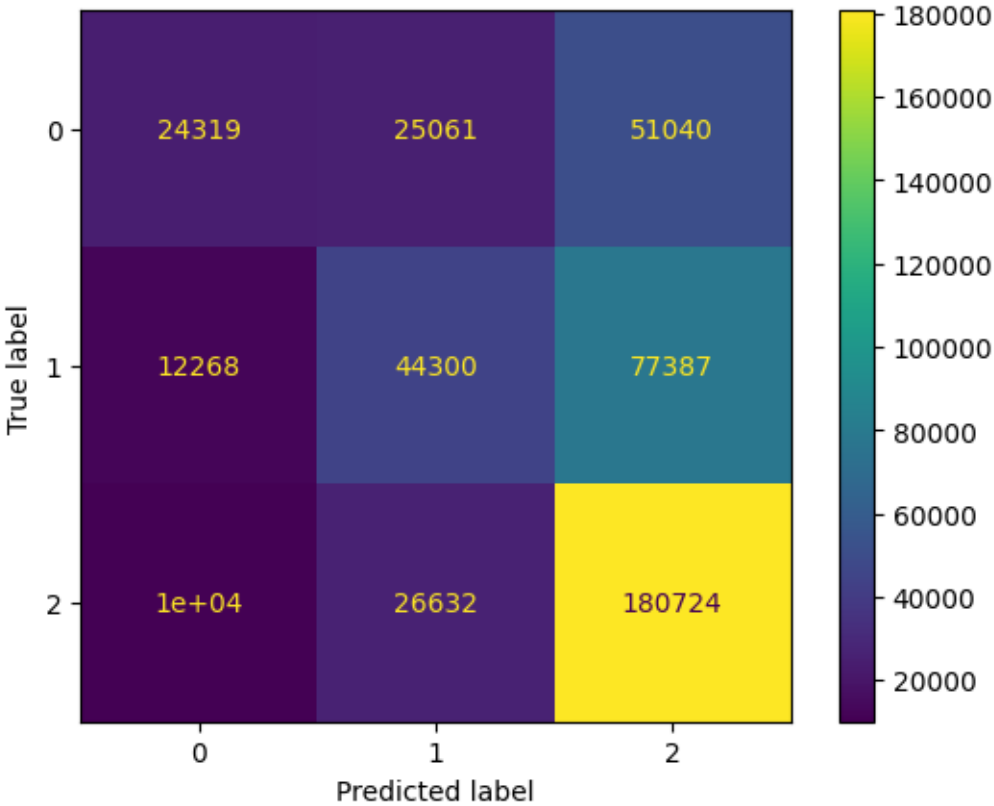
```
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```



```
Accuracy: 0.54273
Recall: [0.21933878 0.3227875  0.82761967]
Precision: [0.51603683 0.44834201 0.57548464]
F-1 score: [0.30783422 0.37534343 0.67889808]
====================

stemming=True, normalize=True, add_sentiment=False, add_location=False,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(

====================
```

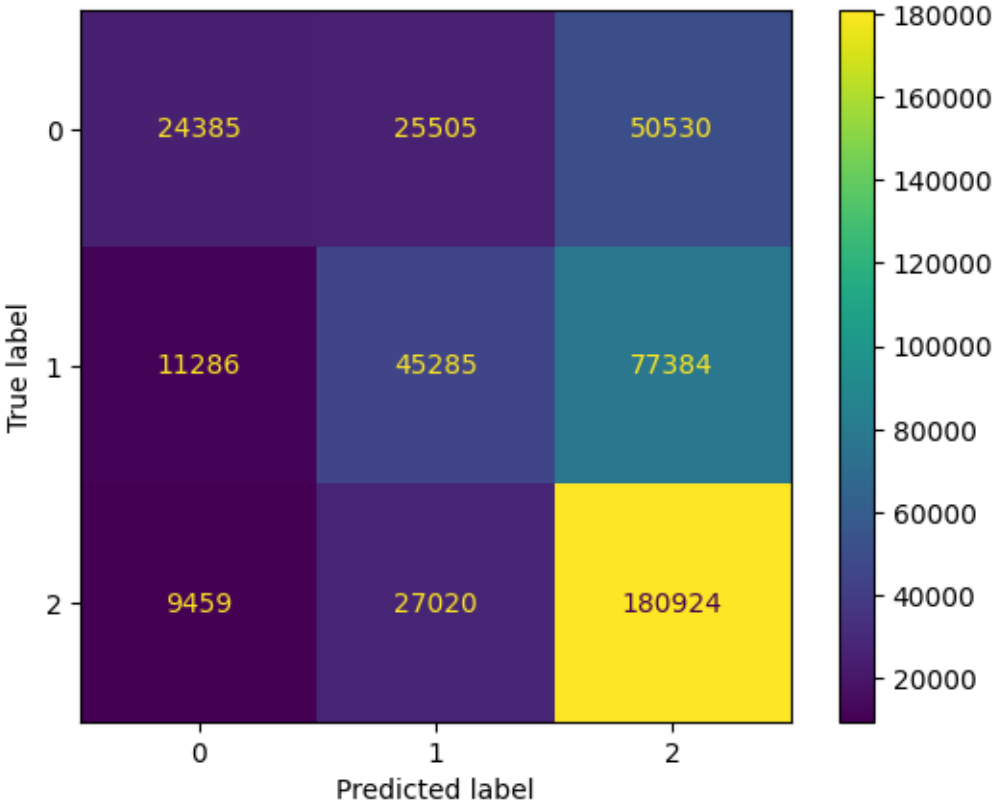


```
Accuracy: 0.54665
Recall: [0.22391954 0.33588145 0.82558198]
Precision: [0.5144596  0.4519593  0.58175996]
F-1 score: [0.3120282  0.38536911 0.68254988]
====================

stemming=False, normalize=True, add_sentiment=True, add_location=True,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```



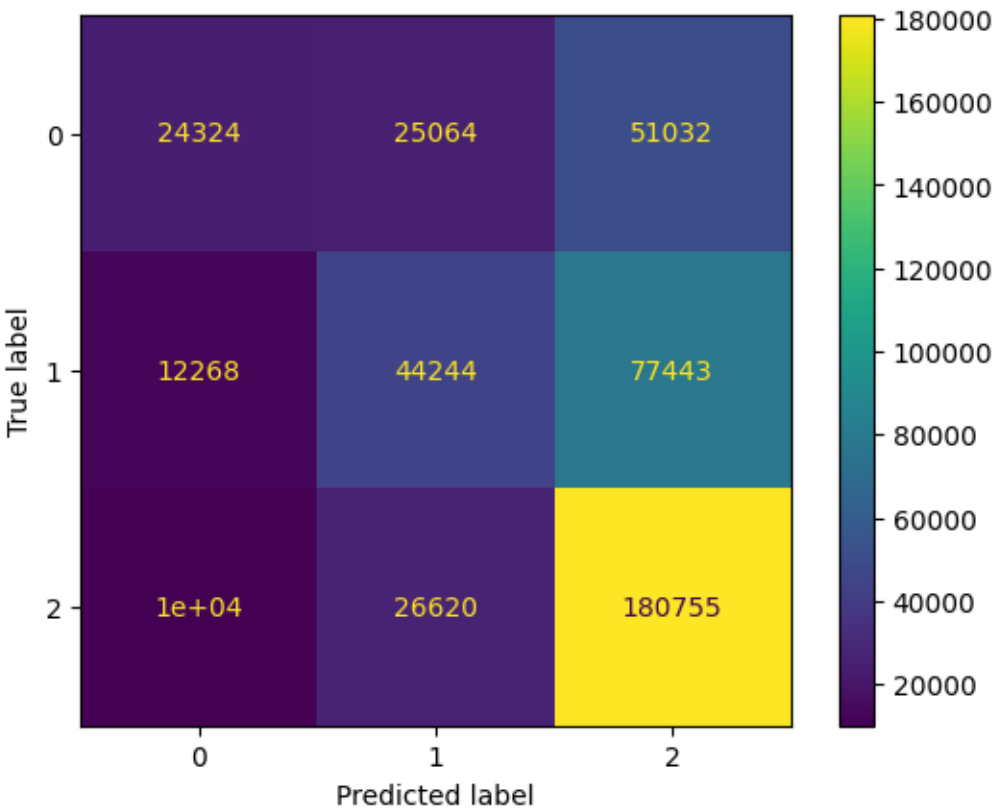

```
Accuracy: 0.55191
Recall: [0.24217287 0.33070807 0.83128568]
Precision: [0.52148647 0.46149198 0.58458164]
F-1 score: [0.33074925 0.3853045  0.68644052]
====================

stemming=False, normalize=True, add_sentiment=True, add_location=False,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
  test_mul["full_text_processed"] = pool.starmap(
====================
```




```
Accuracy: 0.55468
Recall: [0.24283011 0.33806129 0.83220563]
Precision: [0.54032794 0.46298947 0.58582169]
F-1 score: [0.33507386 0.39078377 0.68760891]
====================

stemming=False, normalize=True, add_sentiment=False, add_location=True,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
```

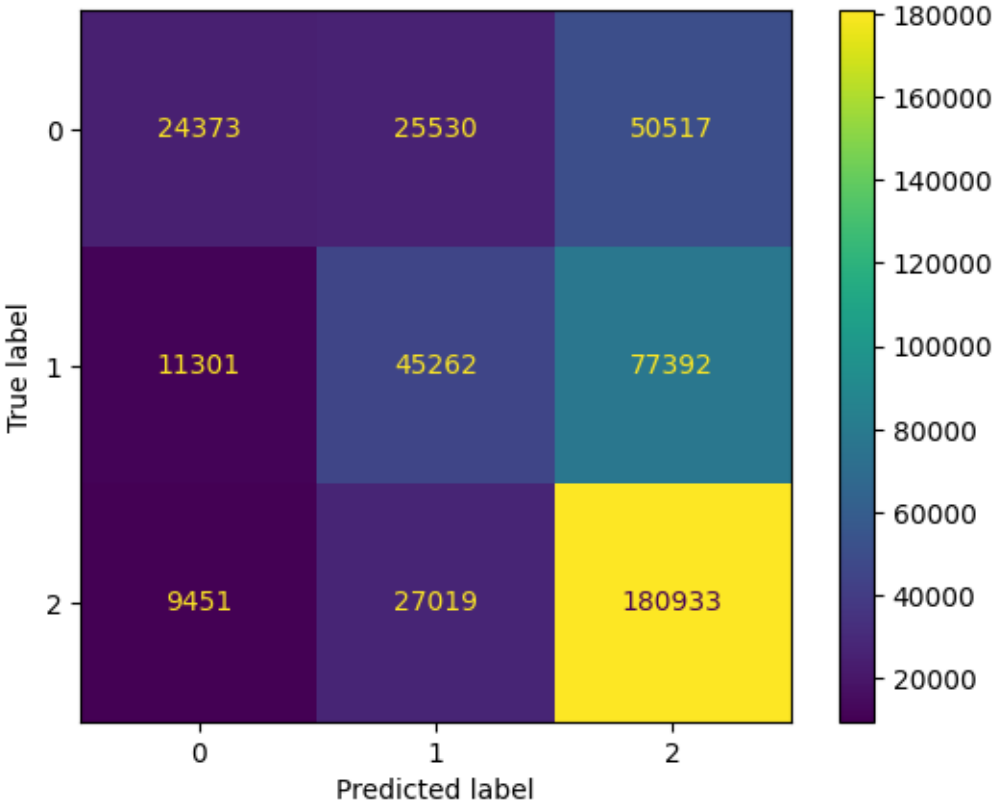
```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```



```
Accuracy: 0.55187
Recall: [0.24222266 0.33029002 0.83142827]
Precision: [0.52175032 0.46122092 0.58453255]
F-1 score: [0.33084875 0.38492625 0.68645527]
====================

stemming=False, normalize=True, add_sentiment=False, add_location=False,
dimension_reduction='svd'
```

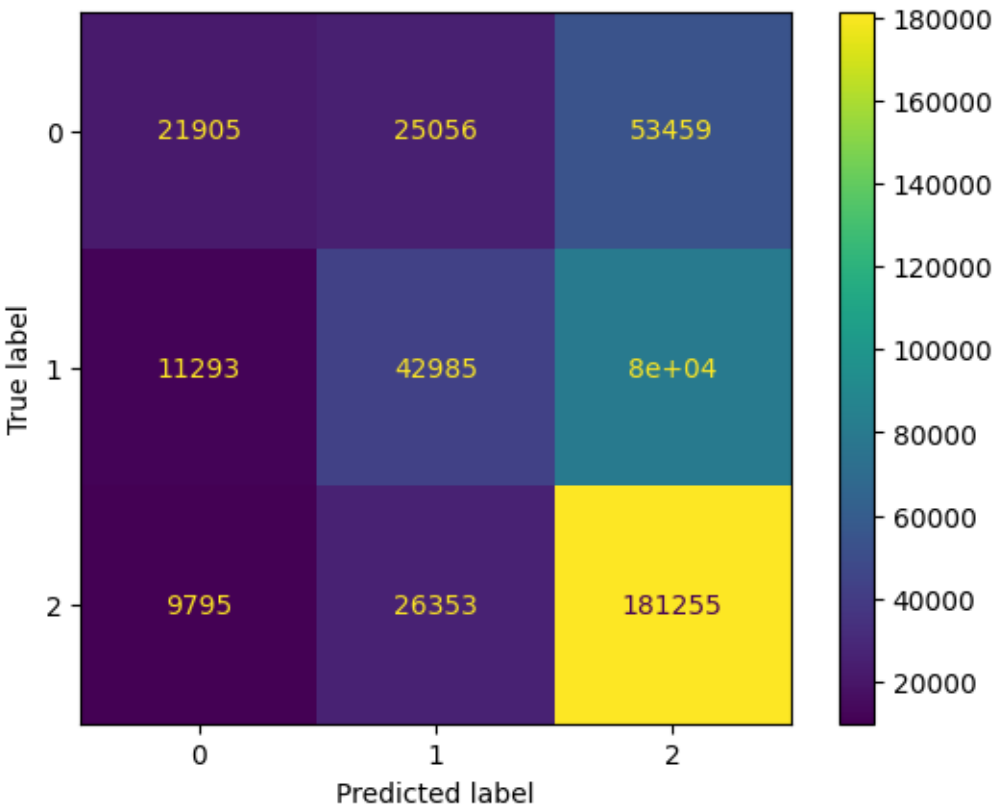
```
/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
```

```
====================
```



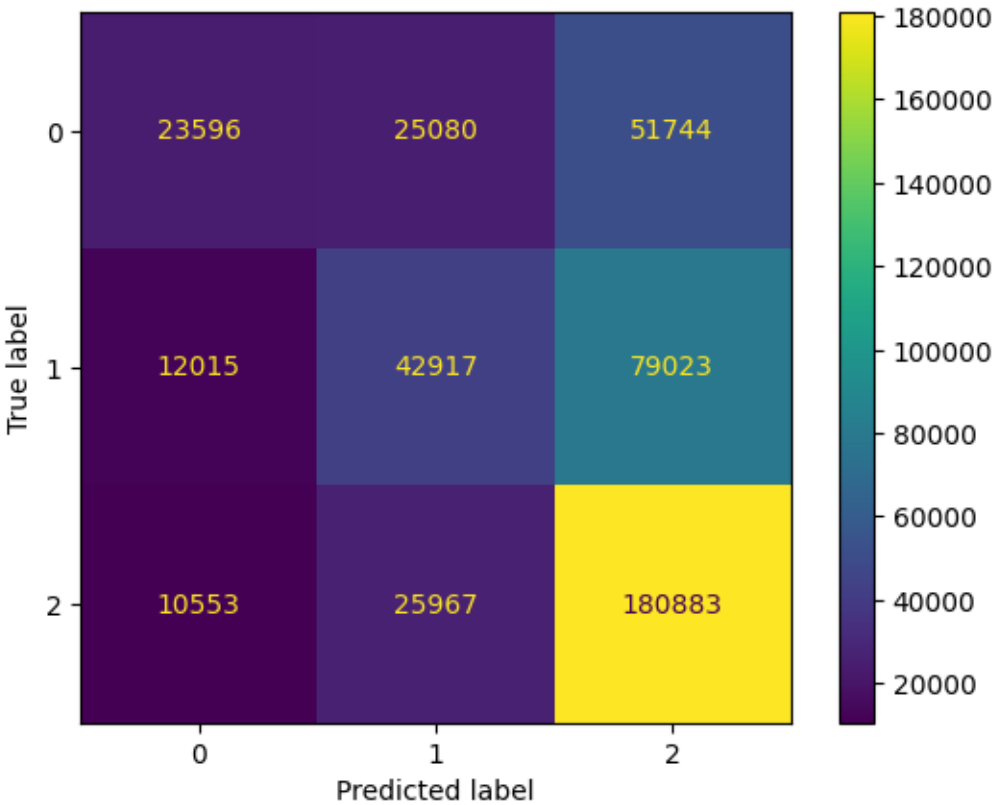

```
Accuracy: 0.55463
Recall: [0.24271062 0.33788959 0.83224703]
Precision: [0.54012188 0.46274959 0.58584325]
F-1 score: [0.33492047 0.39058361 0.68763789]
====================

stemming=True, normalize=False, add_sentiment=True, add_location=True,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```

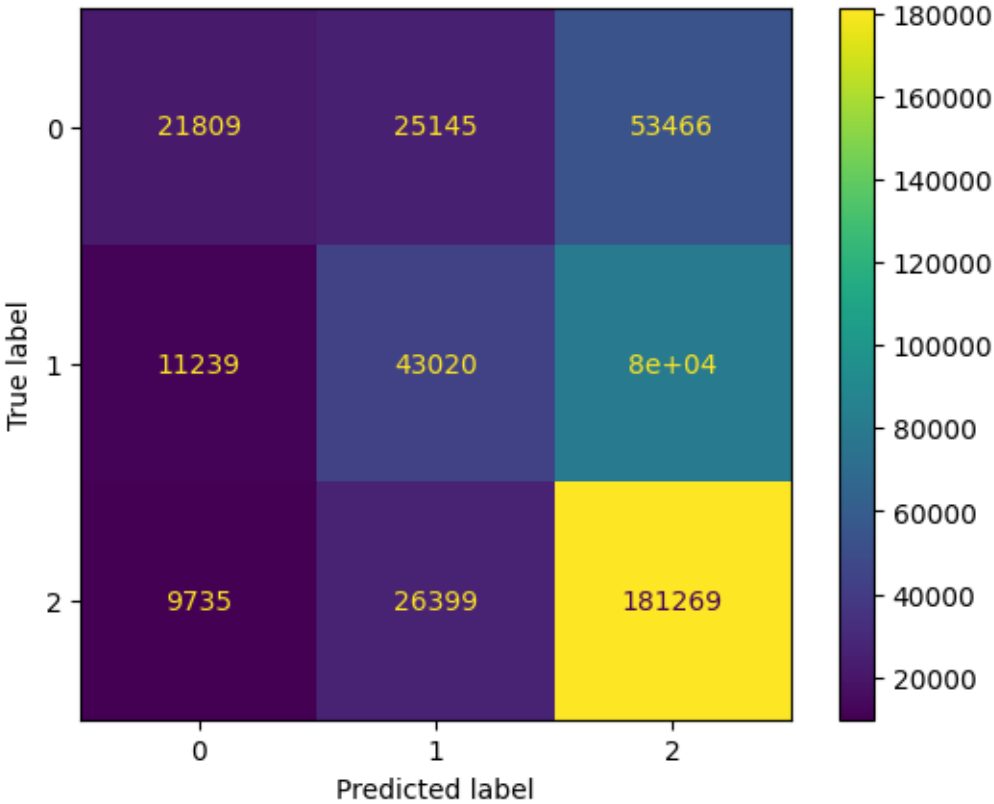



```
Accuracy: 0.54484
Recall: [0.21813384 0.32089134 0.83372815]
Precision: [0.50950155 0.45537852 0.57652732]
F-1 score: [0.30548137 0.37648512 0.68167373]
```

```
====================

stemming=True, normalize=False, add_sentiment=True, add_location=False,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(

====================
```

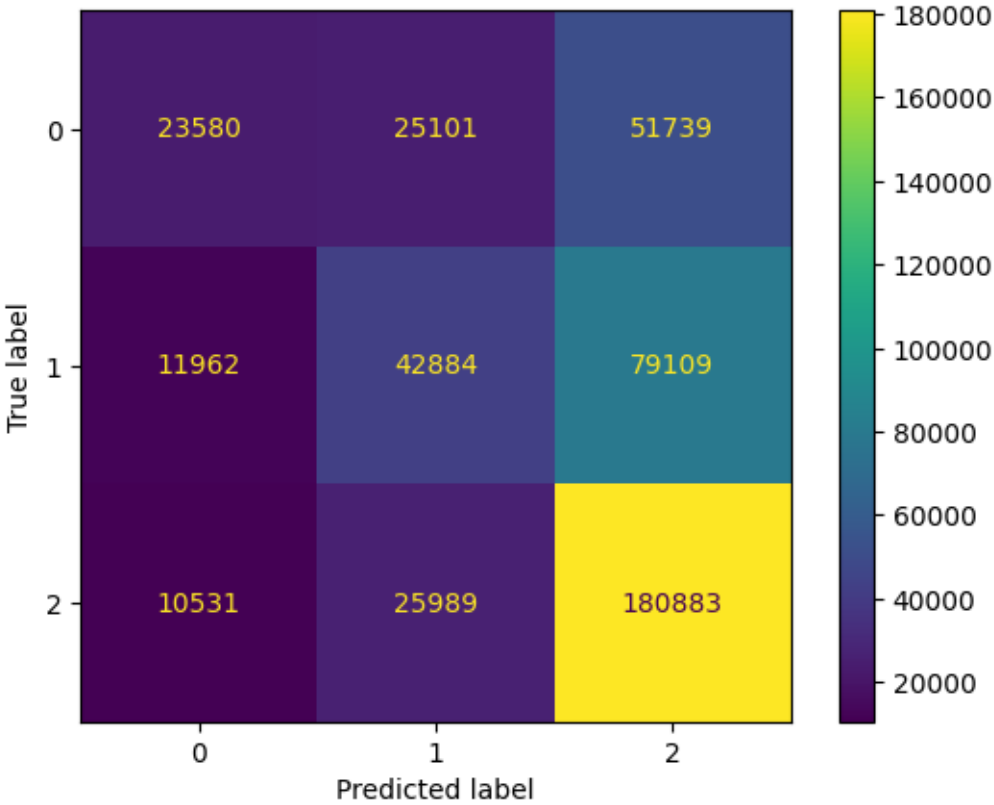


```
Accuracy: 0.54761
Recall: [0.23497311 0.32038371 0.83201704]
Precision: [0.51113422 0.45673875 0.5804043 ]
F-1 score: [0.3219451  0.3765987  0.68379917]
====================

stemming=True, normalize=False, add_sentiment=False, add_location=True,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```



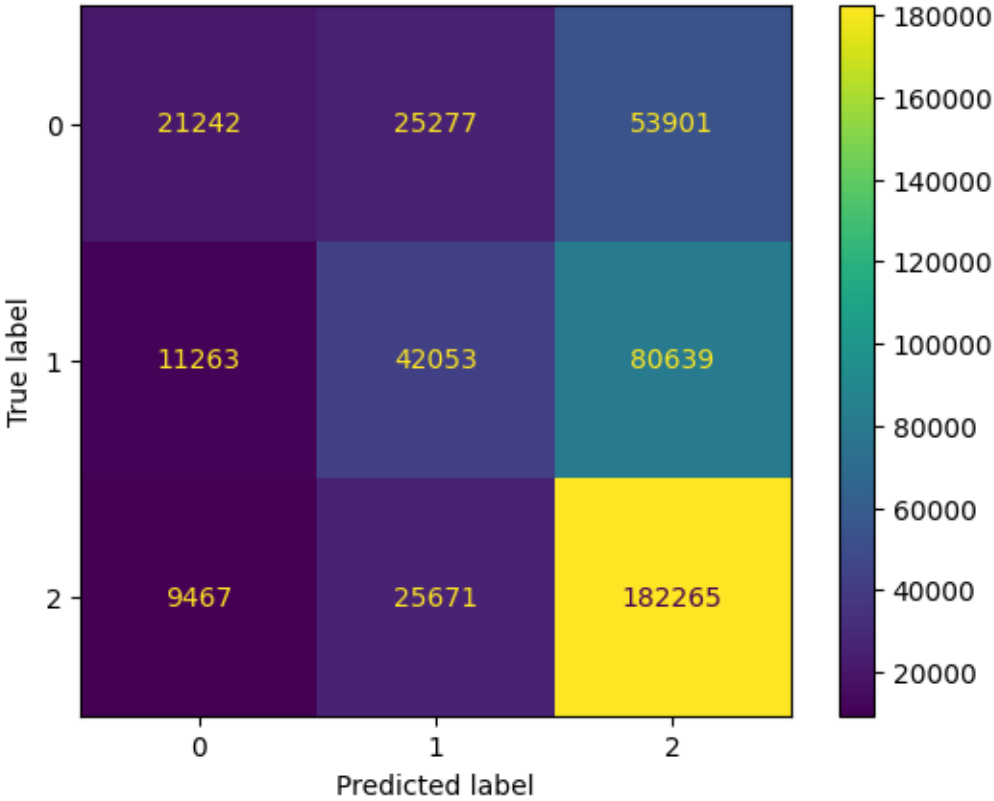

```
Accuracy: 0.54473
Recall: [0.21717785 0.32115263 0.83379254]
Precision: [0.50975855 0.45492999 0.5764985 ]
F-1 score: [0.30458859 0.37651136 0.68167511]
====================

stemming=True, normalize=False, add_sentiment=False, add_location=False,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
  test_mul["full_text_processed"] = pool.starmap(
====================
```

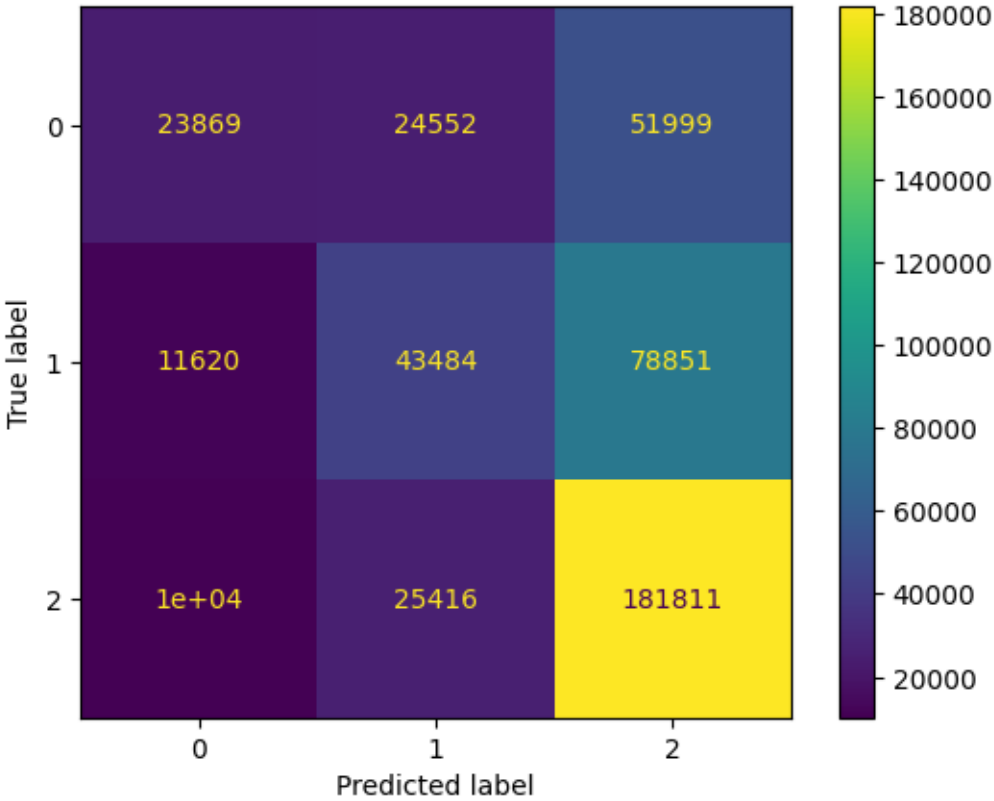



```
Accuracy: 0.5475
Recall: [0.23481378 0.32013736 0.83201704]
Precision: [0.5117965  0.45633899 0.58025349]
F-1 score: [0.32192664 0.37629262 0.68369449]
====================

stemming=False, normalize=False, add_sentiment=True, add_location=True,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
```

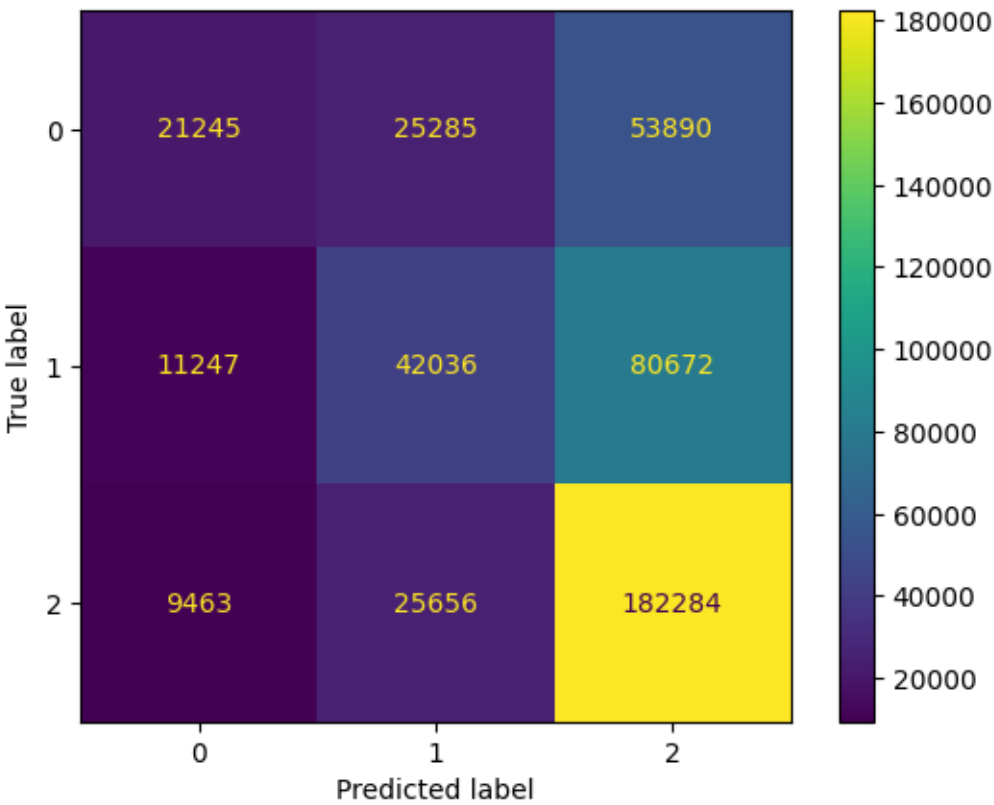
```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```



```
Accuracy: 0.54354
Recall: [0.21153157 0.31393378 0.8383739 ]
Precision: [0.5060993  0.45217793 0.57532236]
F-1 score: [0.29835946 0.37058284 0.68237466]
====================

stemming=False, normalize=False, add_sentiment=True, add_location=False,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(

====================
```

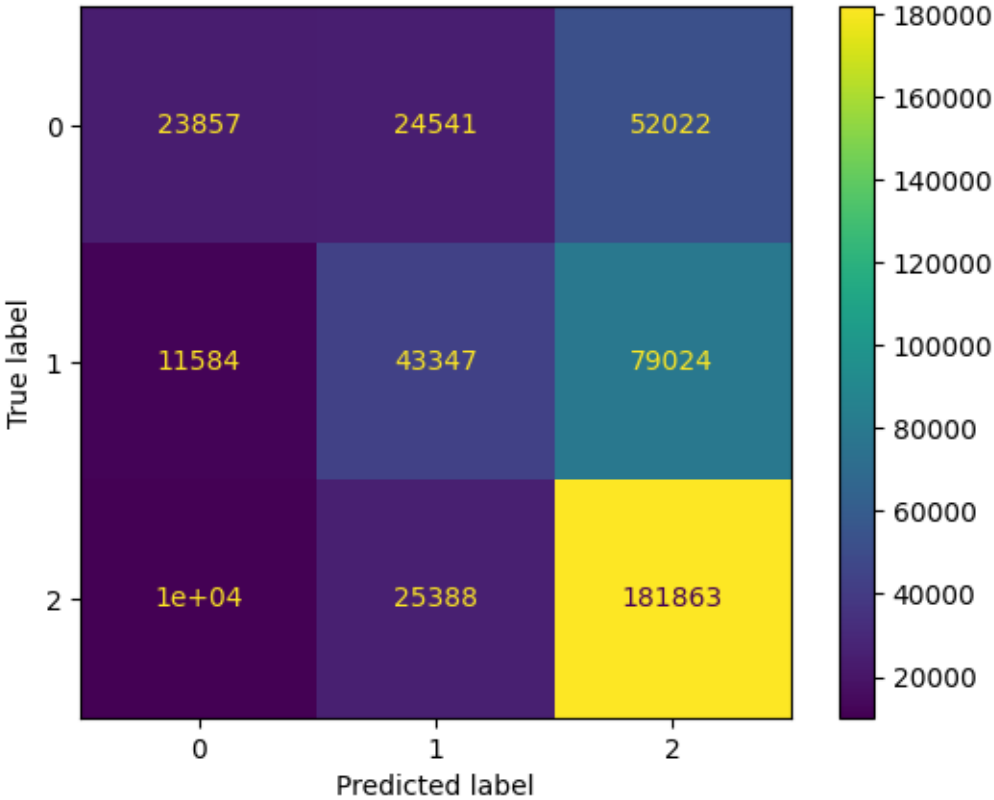


```
Accuracy: 0.55152
Recall: [0.23769169 0.32461648 0.83628561]
Precision: [0.52269791 0.46530839 0.58149561]
F-1 score: [0.32678235 0.38243326 0.68599641]
====================

stemming=False, normalize=False, add_sentiment=False, add_location=True,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```




```
Accuracy: 0.54355
Recall: [0.21156144 0.31380688 0.83846129]
Precision: [0.50637588 0.45211181 0.57530788]
F-1 score: [0.29843723 0.37047221 0.68239342]
====================

stemming=False, normalize=False, add_sentiment=False, add_location=False,
dimension_reduction='svd'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

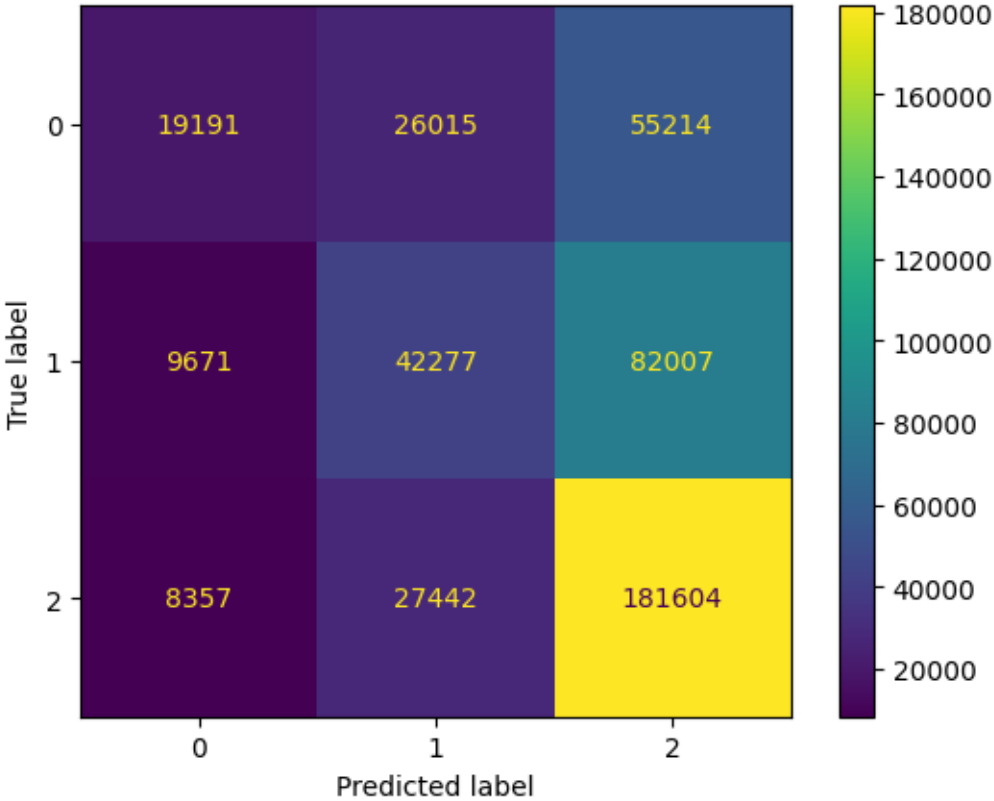
```
  test_mul["full_text_processed"] = pool.starmap(
====================
```



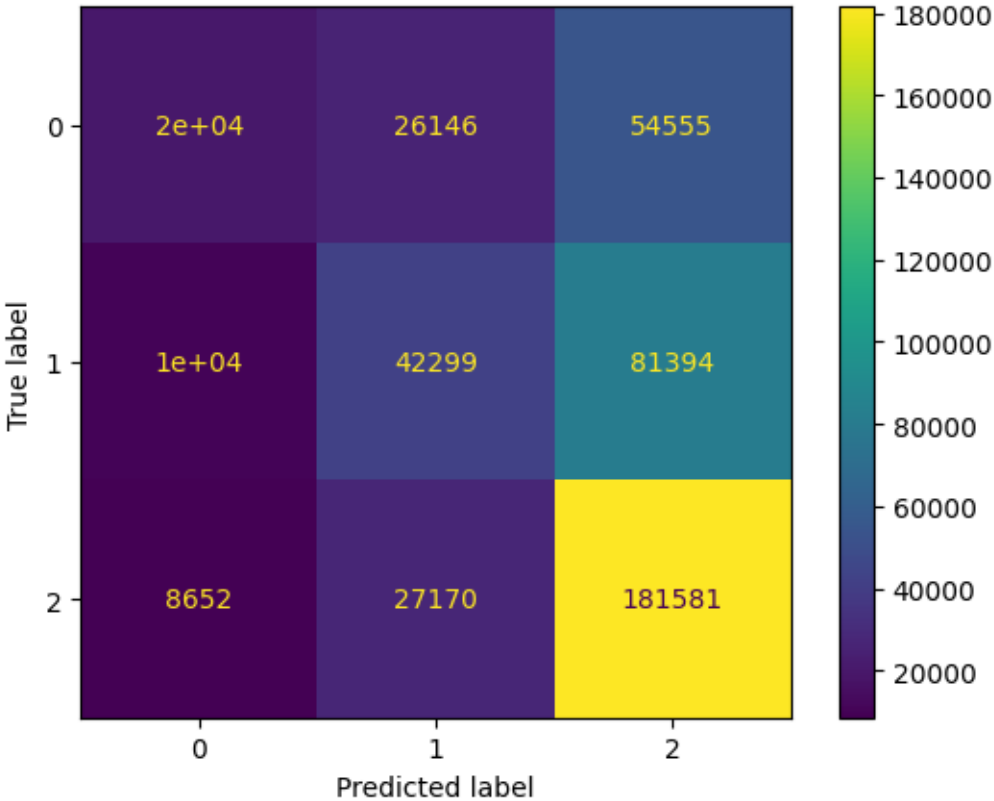

```
Accuracy: 0.5513
Recall: [0.2375722  0.32359374 0.83652479]
Precision: [0.52326015 0.46471761 0.58120092]
F-1 score: [0.32677912 0.38152365 0.68587171]
====================

stemming=True, normalize=True, add_sentiment=True, add_location=True,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

==========



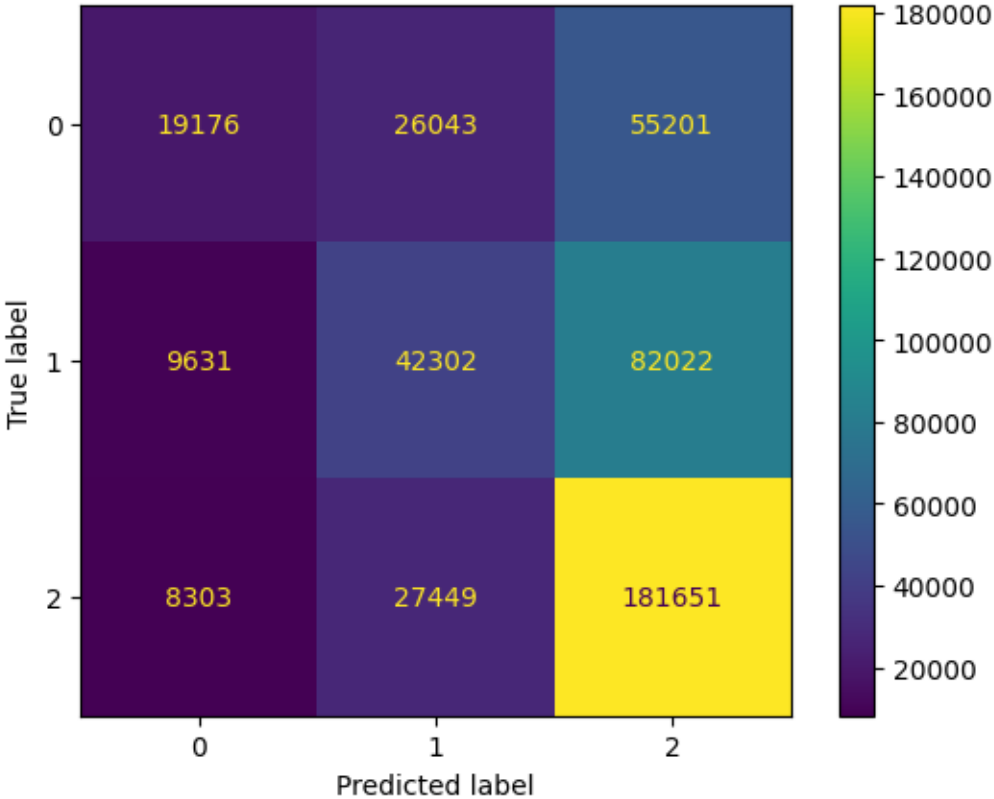
```
Accuracy: 0.53803
Recall: [0.19110735 0.31560599 0.83533346]
Precision: [0.51562374 0.44160904 0.56960401]
F-1 score: [0.27885992 0.36812385 0.67733874]
====================

stemming=True, normalize=True, add_sentiment=True, add_location=False,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(

====================
```




```
Accuracy: 0.5392
Recall: [0.19636527 0.31577022 0.83522766]
Precision: [0.51041855 0.44238875 0.57185463]
F-1 score: [0.28361848 0.36850634 0.6788925 ]
====================

stemming=True, normalize=True, add_sentiment=False, add_location=True,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

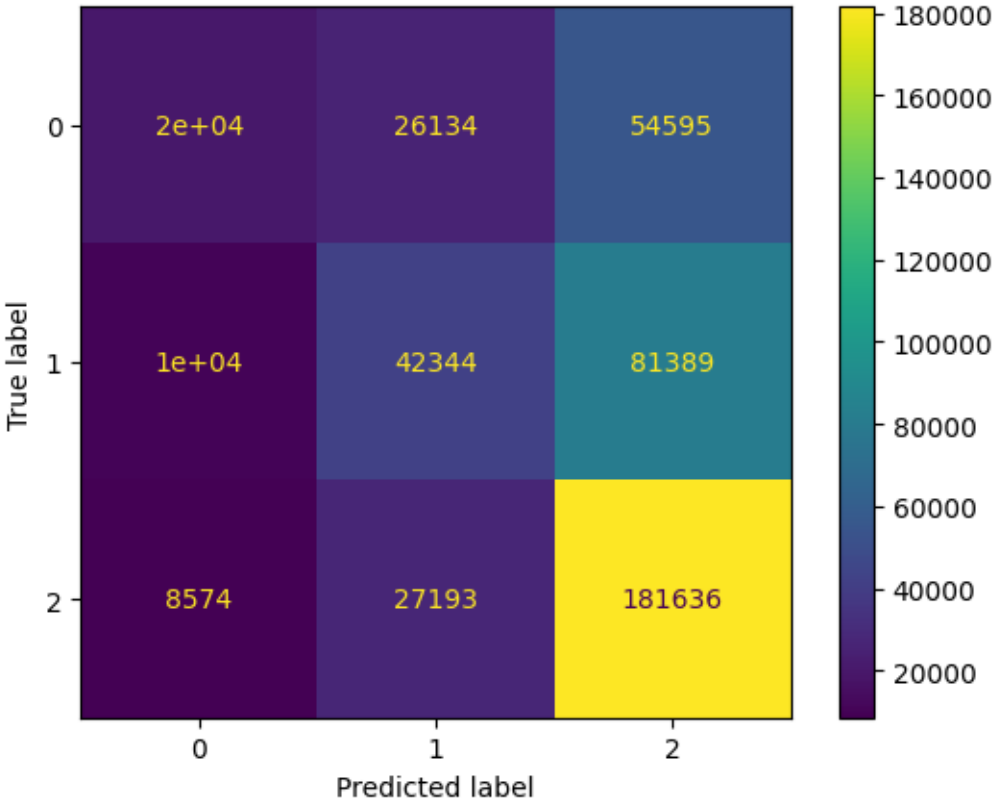
==========



```
Accuracy: 0.53816
Recall: [0.19095798 0.31579262 0.83554965]
Precision: [0.51673403 0.44159342 0.56966388]
F-1 score: [0.27886279 0.36824535 0.67745214]
====================

stemming=True, normalize=True, add_sentiment=False, add_location=False,
dimension_reduction='nmf'
```

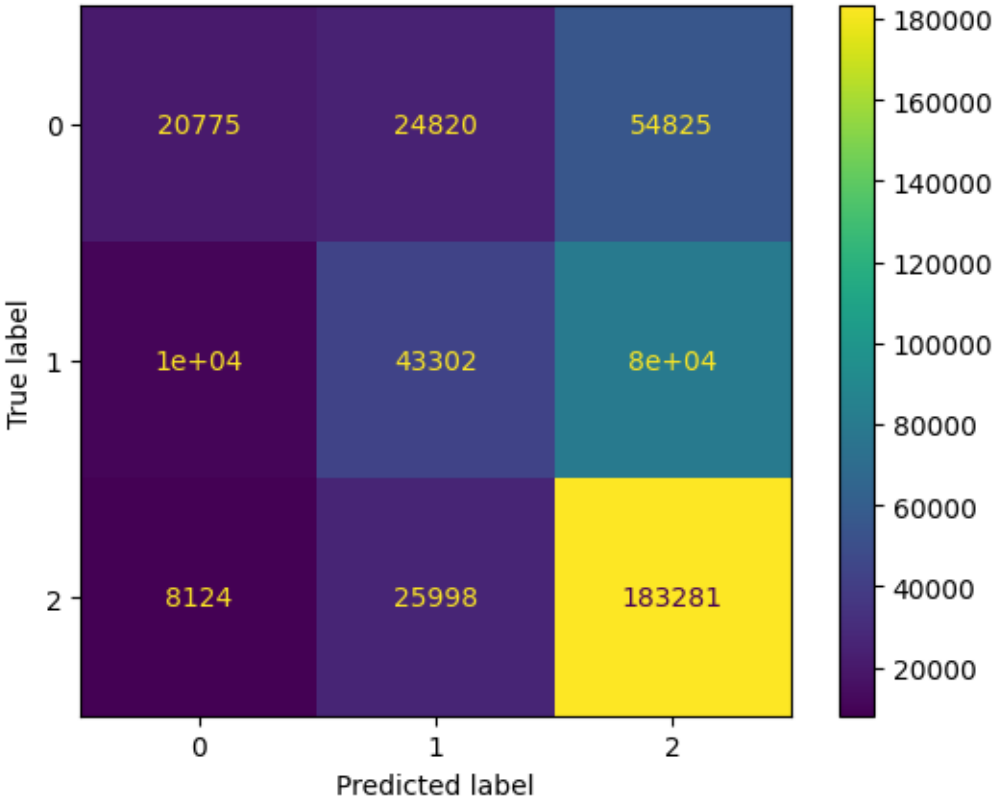
```
/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

```
====================
```




```
Accuracy: 0.53936
Recall: [0.19608644 0.31610616 0.83548065]
Precision: [0.5116273  0.44260016 0.57186575]
F-1 score: [0.28351343 0.36880841 0.67898389]
====================

stemming=False, normalize=True, add_sentiment=True, add_location=True,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

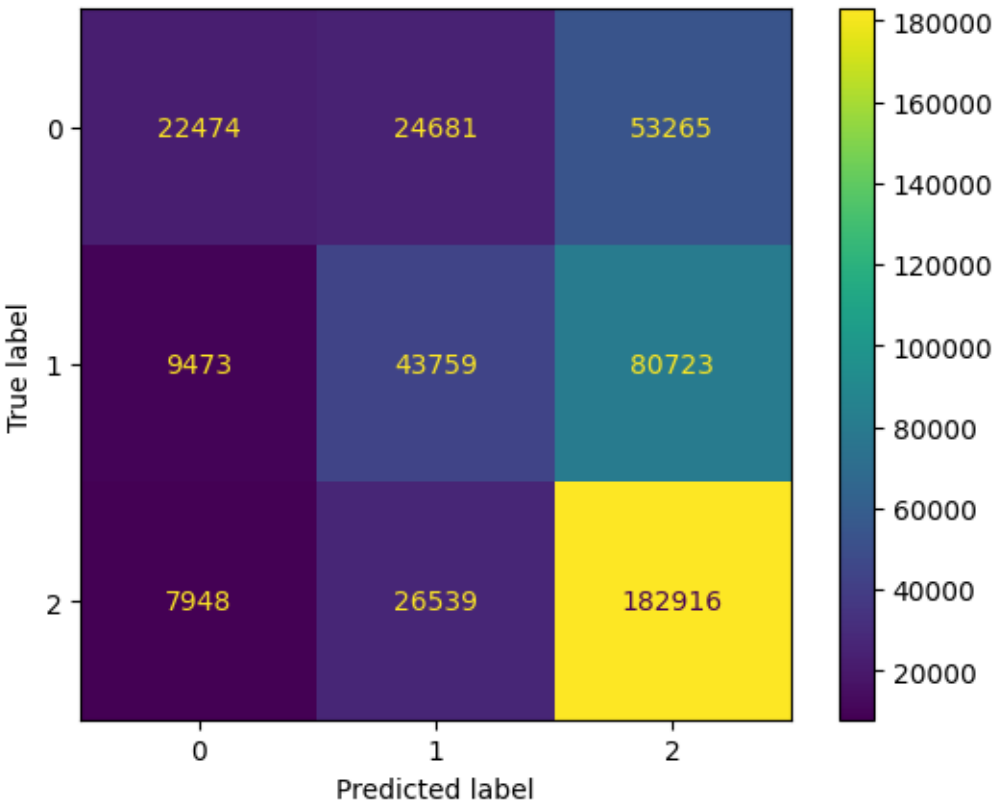
```
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
====================
```



```
Accuracy: 0.54752
Recall: [0.2068811  0.32325781 0.84304724]
Precision: [0.52851837 0.46007225 0.57572169]
F-1 score: [0.29736345 0.3797172  0.68419962]
====================

stemming=False, normalize=True, add_sentiment=True, add_location=False,
dimension_reduction='nmf'
```

```
/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

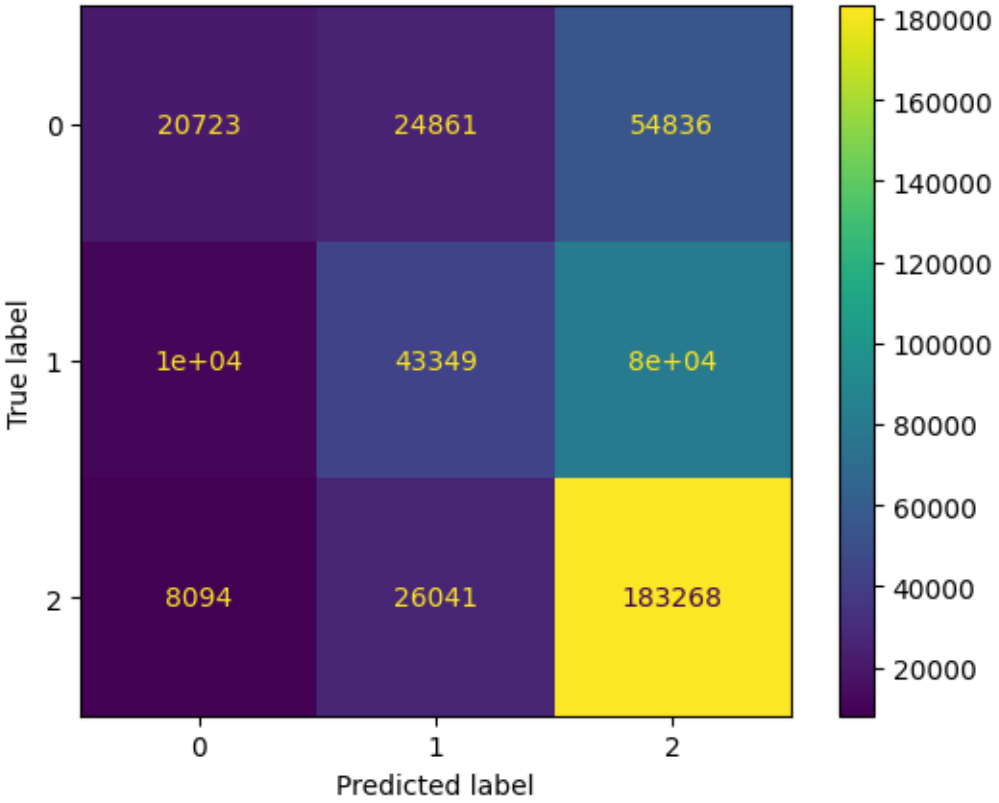
```
====================
```




```
Accuracy: 0.55149
Recall: [0.22380004 0.3266694  0.84136833]
Precision: [0.56332874 0.4607229  0.57719688]
F-1 score: [0.32033639 0.38228485 0.68468502]
====================

stemming=False, normalize=True, add_sentiment=False, add_location=True,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

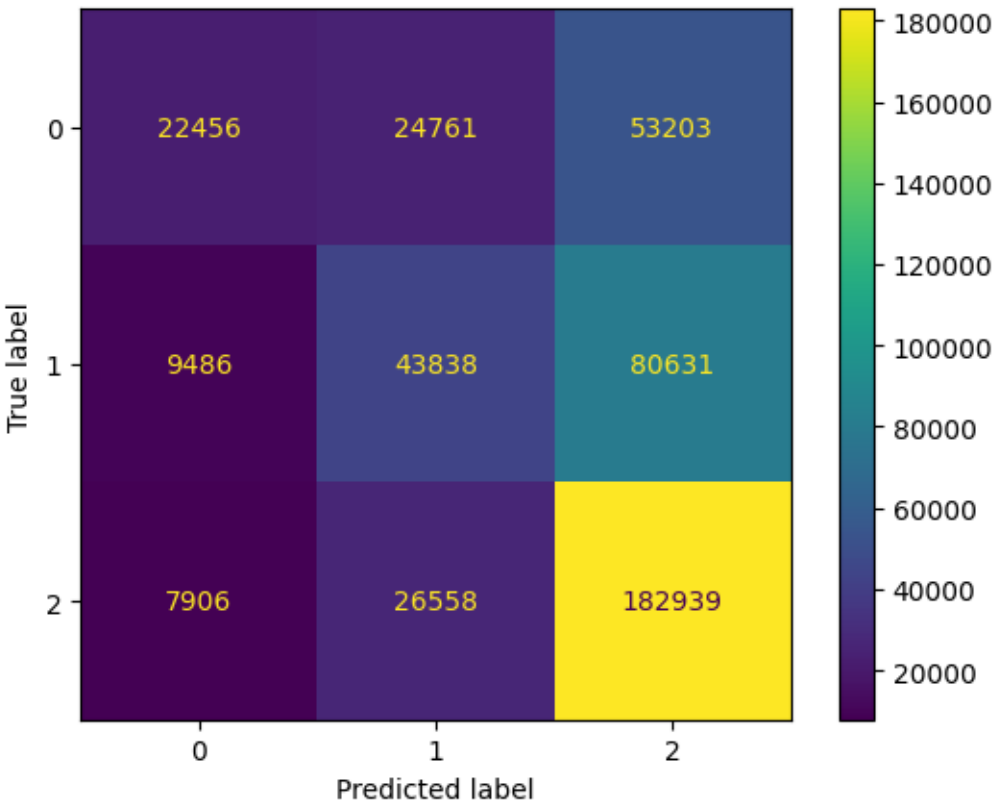
==========



```
Accuracy: 0.54748
Recall: [0.20636327 0.32360867 0.84298745]
Precision: [0.52906635 0.45993146 0.57566639]
F-1 score: [0.29691451 0.37991113 0.68414088]
====================

stemming=False, normalize=True, add_sentiment=False, add_location=False,
dimension_reduction='nmf'
```

```
/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

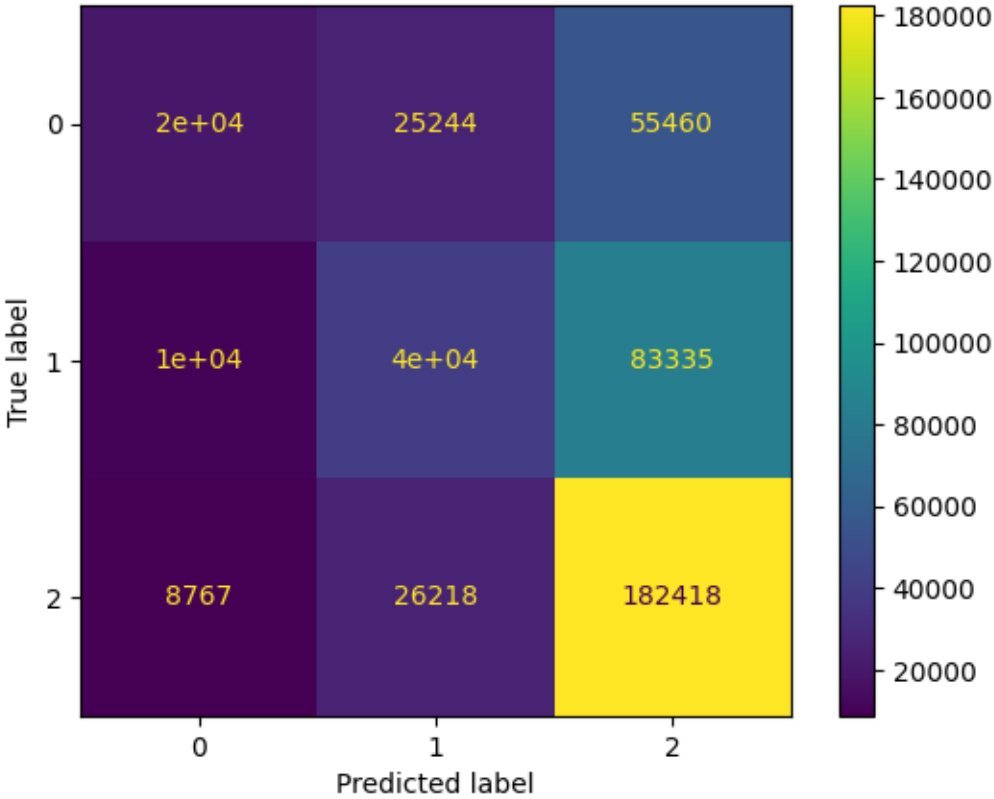
```
====================
```




```
Accuracy: 0.55167
Recall: [0.22362079 0.32725915 0.84147413]
Precision: [0.56354146 0.46069128 0.57750818]
F-1 score: [0.32018707 0.38267747 0.68493905]
====================

stemming=True, normalize=False, add_sentiment=True, add_location=True,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

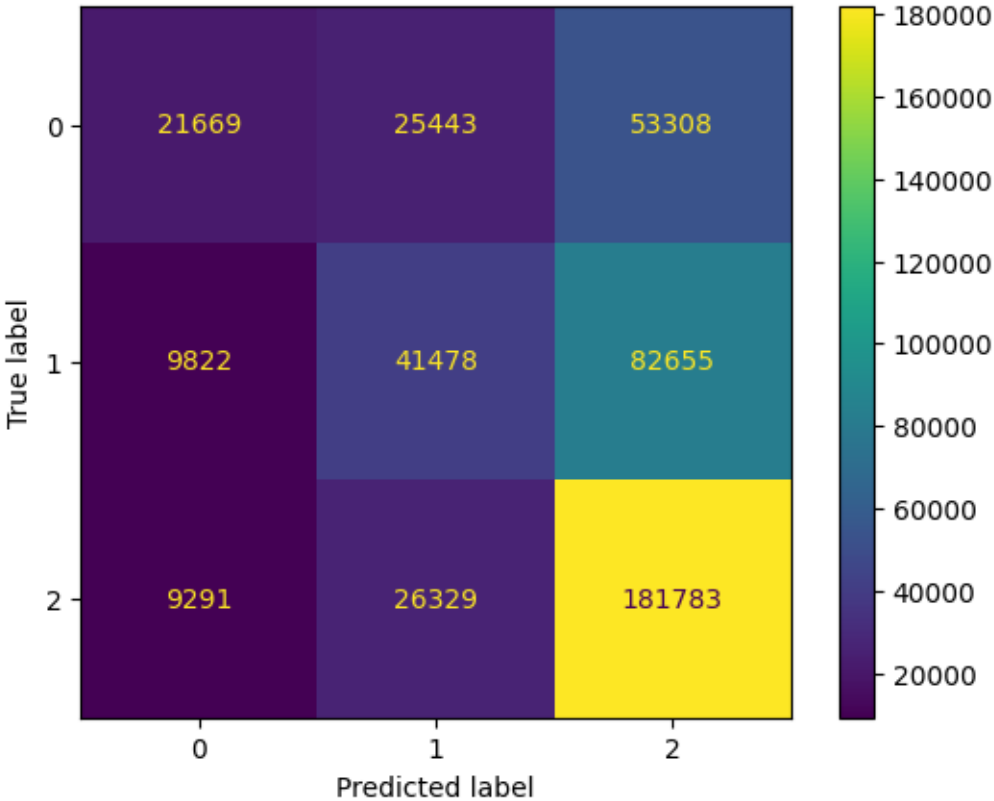
==========



```
Accuracy: 0.53678
Recall: [0.19633539 0.30139226 0.83907766]
Precision: [0.50906274 0.43962542 0.56790354]
F-1 score: [0.28337765 0.35761548 0.67735827]
====================

stemming=True, normalize=False, add_sentiment=True, add_location=False,
dimension_reduction='nmf'
```

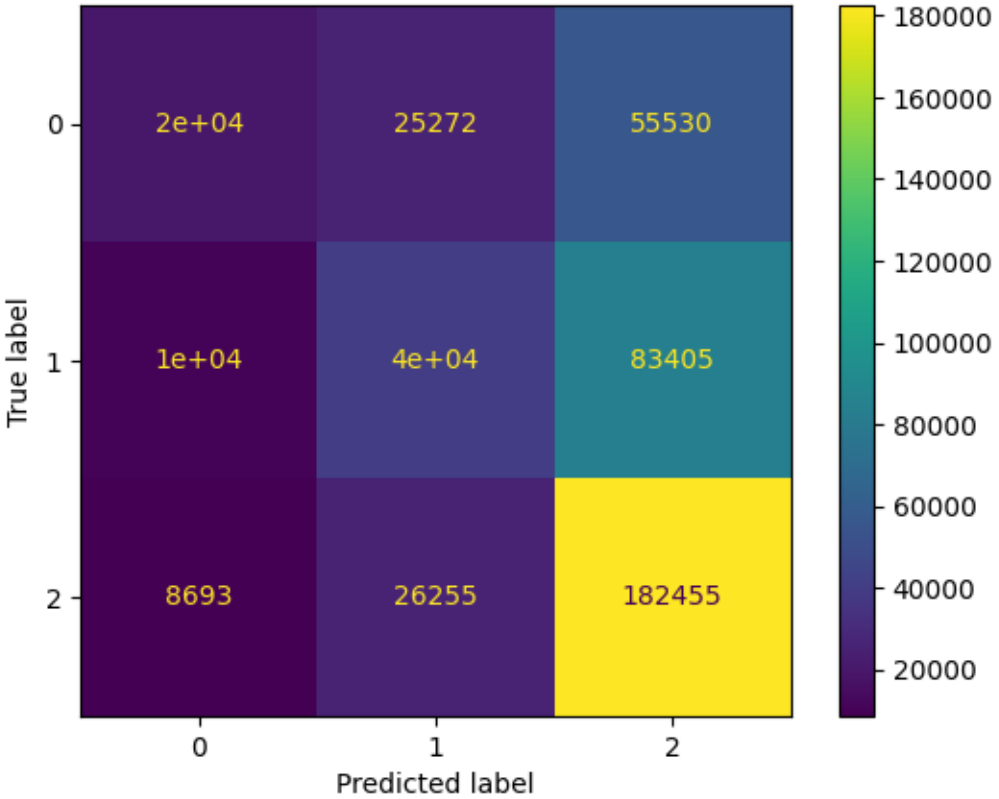
```
/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

```
====================
```




```
Accuracy: 0.54215
Recall: [0.21578371 0.3096413  0.83615681]
Precision: [0.53133735 0.44480429 0.57210162]
F-1 score: [0.306922   0.3651152  0.67937341]
====================

stemming=True, normalize=False, add_sentiment=False, add_location=True,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

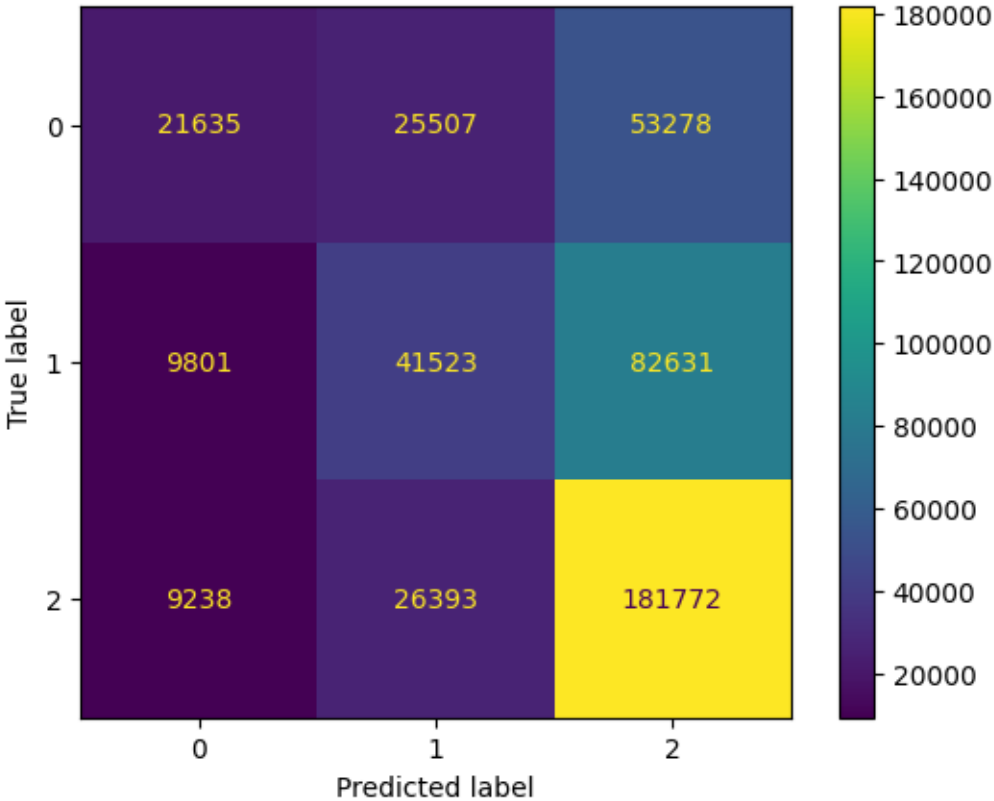
```
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
====================
```



```
Accuracy: 0.53668
Recall: [0.19535949 0.30148184 0.83924785]
Precision: [0.50987629 0.43938768 0.5677059 ]
F-1 score: [0.28248474 0.35759983 0.67727309]
====================

stemming=True, normalize=False, add_sentiment=False, add_location=False,
dimension_reduction='nmf'
```

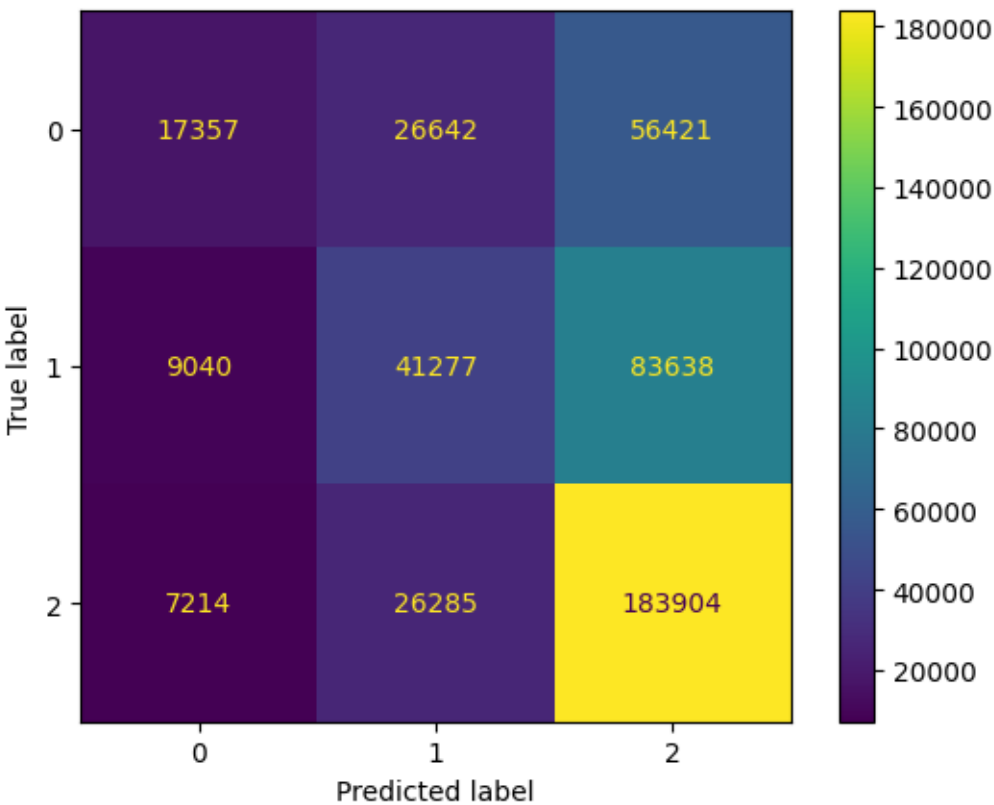
```
/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
```

```
====================
```



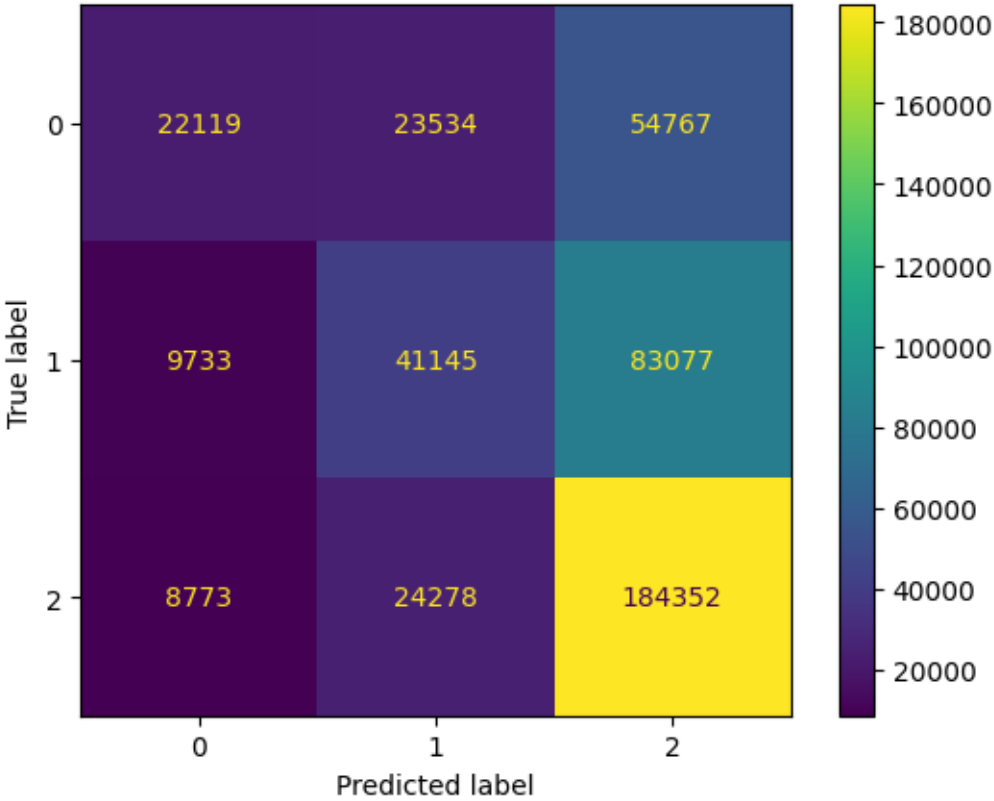

```
Accuracy: 0.54215
Recall: [0.21544513 0.30997723 0.83610622]
Precision: [0.53191228 0.44446228 0.57218405]
F-1 score: [0.30667498 0.36523322 0.67941482]
====================

stemming=False, normalize=False, add_sentiment=True, add_location=True,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```

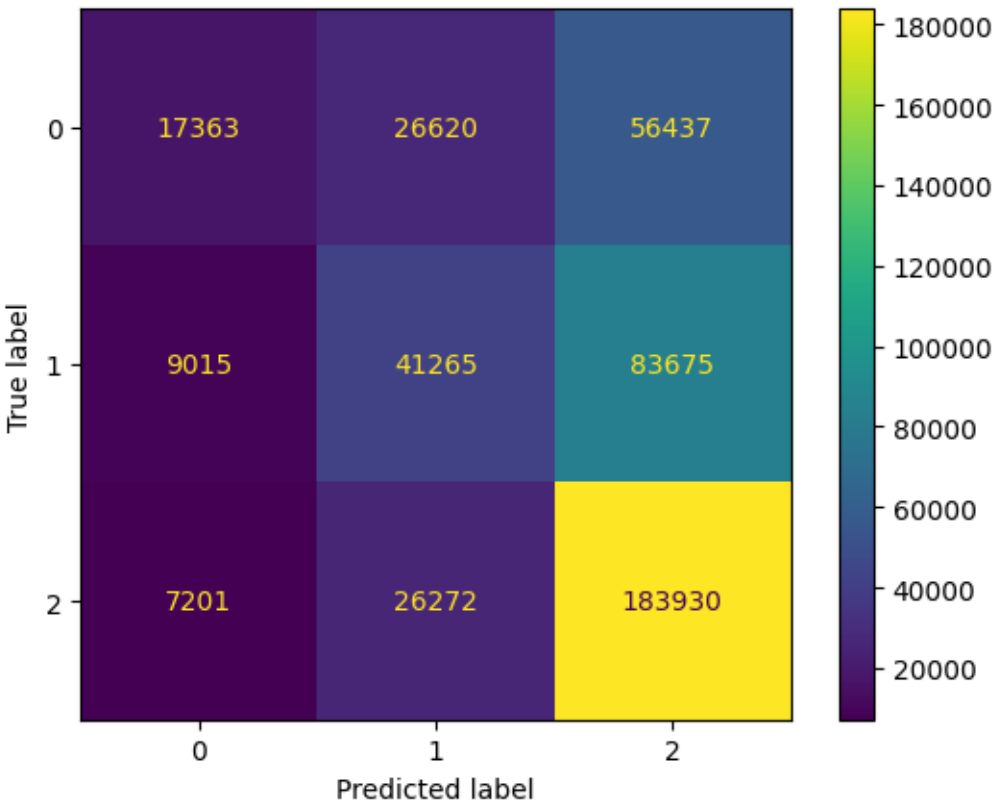



```
Accuracy: 0.53685
Recall: [0.17284405 0.30814079 0.84591289]
Precision: [0.51640832 0.43816611 0.56766976]
F-1 score: [0.25899978 0.36182662 0.67940728]
```

```
====================

stemming=False, normalize=False, add_sentiment=True, add_location=False,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(

====================
```

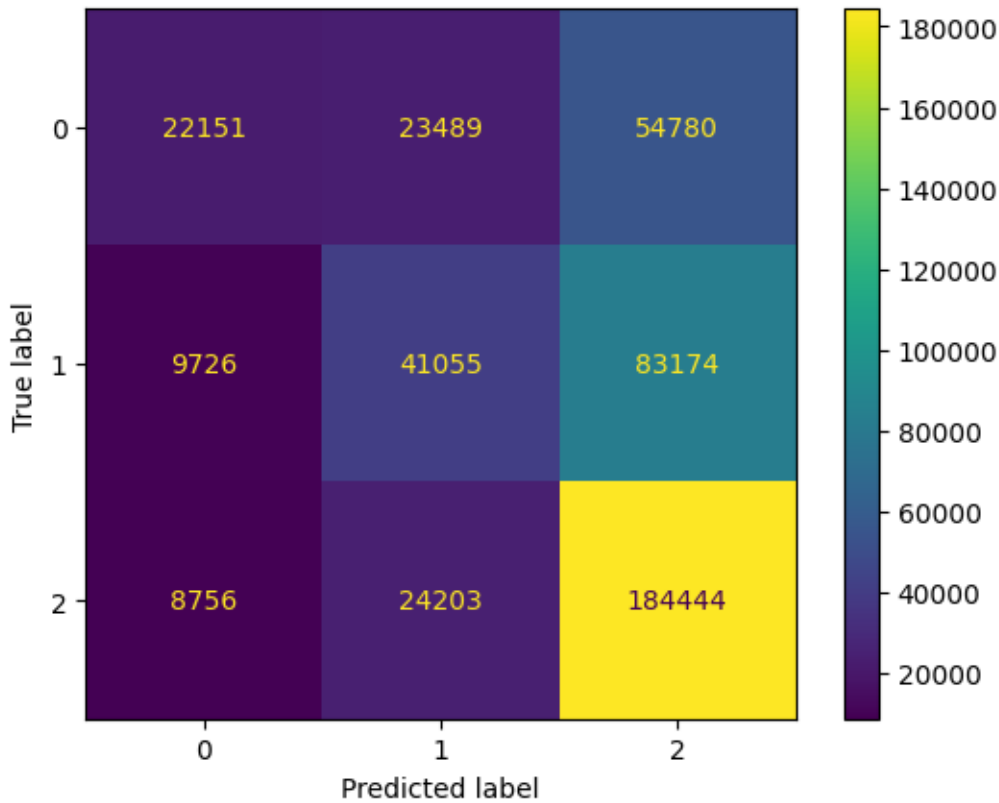


```
Accuracy: 0.54809
Recall: [0.22026489 0.30715539 0.84797358]
Precision: [0.54446769 0.46252684 0.57217346]
F-1 score: [0.31364458 0.36915913 0.68329259]
====================

stemming=False, normalize=False, add_sentiment=False, add_location=True,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:17: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] += " " + train_mul["location"]
/tmp/ipykernel_7394/588642200.py:18: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  test_mul["full_text_processed"] += " " + test_mul["location"]

====================
```




```
Accuracy: 0.5369
Recall: [0.1729038  0.30805121 0.84603248]
Precision: [0.51707913 0.43825738 0.56761161]
F-1 score: [0.25915119 0.36179596 0.67940419]
====================

stemming=False, normalize=False, add_sentiment=False, add_location=False,
dimension_reduction='nmf'

/tmp/ipykernel_7394/588642200.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  train_mul["full_text_processed"] = pool.starmap(
/tmp/ipykernel_7394/588642200.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
  test_mul["full_text_processed"] = pool.starmap(
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1665: ConvergenceWarning: Maximum number
of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(
====================
```




```
Accuracy: 0.54817
Recall: [0.22058355 0.30648352 0.84839676]
Precision: [0.54514803 0.46260719 0.57210032]
F-1 score: [0.31408052 0.36869898 0.68337776]
====================
```

[14]:
```
{(True, True, True, True, 'svd'): 0.54272,
 (True, True, False, True, 'svd'): 0.54666,
 (True, True, True, False, 'svd'): 0.54273,
 (True, True, False, False, 'svd'): 0.54665,
 (False, True, True, True, 'svd'): 0.55191,
 (False, True, False, True, 'svd'): 0.55468,
 (False, True, True, False, 'svd'): 0.55187,
```

```
 (False, True, False, False, 'svd'): 0.55463,
 (True, False, True, True, 'svd'): 0.54484,
 (True, False, False, True, 'svd'): 0.54761,
 (True, False, True, False, 'svd'): 0.54473,
 (True, False, False, False, 'svd'): 0.5475,
 (False, False, True, True, 'svd'): 0.54354,
 (False, False, False, True, 'svd'): 0.55152,
 (False, False, True, False, 'svd'): 0.54355,
 (False, False, False, False, 'svd'): 0.5513,
 (True, True, True, True, 'nmf'): 0.53803,
 (True, True, False, True, 'nmf'): 0.5392,
 (True, True, True, False, 'nmf'): 0.53816,
 (True, True, False, False, 'nmf'): 0.53936,
 (False, True, True, True, 'nmf'): 0.54752,
 (False, True, False, True, 'nmf'): 0.55149,
 (False, True, True, False, 'nmf'): 0.54748,
 (False, True, False, False, 'nmf'): 0.55167,
 (True, False, True, True, 'nmf'): 0.53678,
 (True, False, False, True, 'nmf'): 0.54215,
 (True, False, True, False, 'nmf'): 0.53668,
 (True, False, False, False, 'nmf'): 0.54215,
 (False, False, True, True, 'nmf'): 0.53685,
 (False, False, False, True, 'nmf'): 0.54809,
 (False, False, True, False, 'nmf'): 0.5369,
 (False, False, False, False, 'nmf'): 0.54817}
```

[15]:
```
(
    best_stemming,
    best_normalize,
    best_add_location,
    best_add_sentiment,
    best_dimension_reduction,
) = max(params2acc, key=params2acc.get)

(
    best_stemming,
    best_normalize,
    best_add_location,
    best_add_sentiment,
    best_dimension_reduction,
)
```

[15]:
```
(False, True, False, True, 'svd')
```

[16]:
```
params2acc[
    (
        best_stemming,
```

```
        best_normalize,
        best_add_location,
        best_add_sentiment,
        best_dimension_reduction,
    )
]
```

[16]: 0.55468

0.0.1 Sentence Transformer (Pre-trained Embeddings)

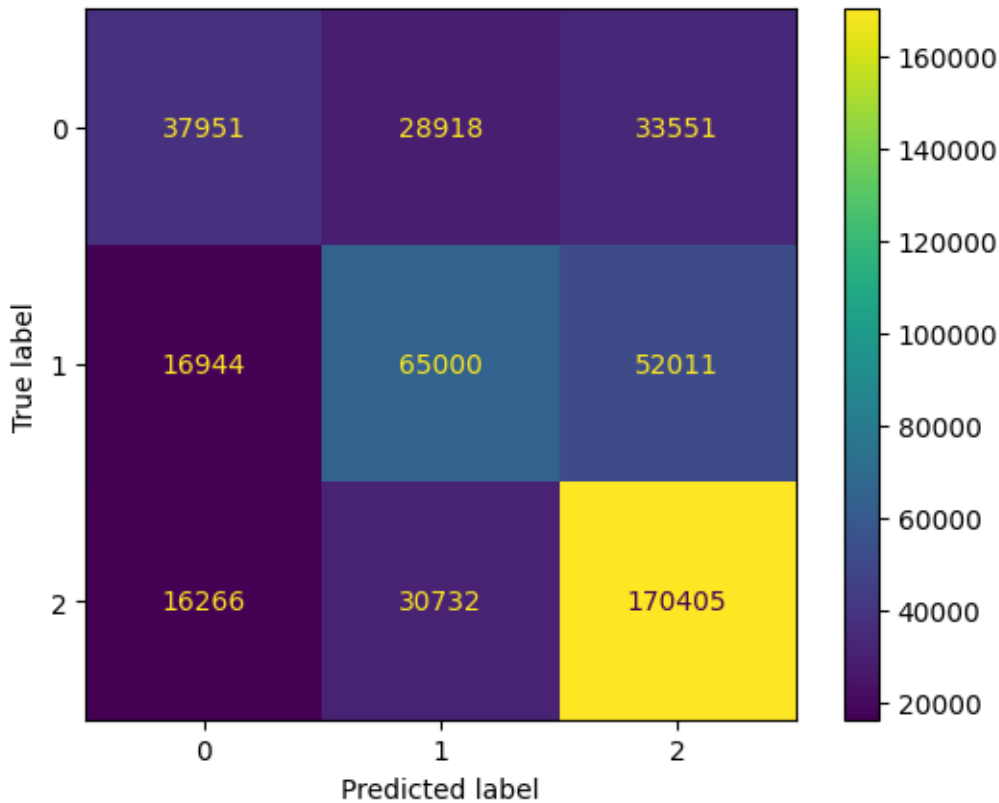
[1]:
```
with Pool(24) as pool:
    data_df["full_text_processed"] = pool.starmap(
        preprocess_text,
        [(t, best_stemming, best_normalize) for t in data_df["full_text"]],
    )

    if best_add_location:
        data_df["full_text_processed"] += " " + data_df["location"]

    if best_add_sentiment:
        sentiments = pool.map(get_sentiment, data_df["full_text_processed"])
```

[18]:
```
model_name = "all-mpnet-base-v2"
model = SentenceTransformer(model_name)
st_embeddings = model.encode(data_df["full_text_processed"],␣
 ↪show_progress_bar=True)
if best_add_sentiment:
    st_embeddings = np.concatenate(
        (st_embeddings, np.expand_dims(np.array(sentiments), 1)), axis=1
    )
train_st_embeddings, test_st_embeddings = (
    st_embeddings[train_idxs], st_embeddings[test_idxs]
)
```

```
Batches:   0%|          | 0/15687 [00:00<?, ?it/s]
```

[19]:
```
one_one_svm = OneVsOneClassifier(LinearSVC(random_state=SEED))
y_pred = one_one_svm.fit(
    train_st_embeddings, train_mul["leading_label_cat"]
).predict(test_st_embeddings)
evaluate_mul(test_mul["leading_label_cat"], y_pred, average=None,␣
 ↪clf=one_one_svm)
```

```
====================
```




```
Accuracy: 0.60507
Recall: [0.37792272 0.48523758 0.78382083]
Precision: [0.53331179 0.52146009 0.66573035]
F-1 score: [0.44236833 0.50269716 0.71996535]
====================
```

[19]: 0.60507

0.0.2 Fine-tune Sequence-Classifying Transfomer

[2]:
```
import torch
from transformers import AutoTokenizer


train_idxs_array = np.array(train_idxs)
train_idxs, valid_idxs = get_train_test_indices(len(train_idxs), test_size=0.2)
train_idxs, valid_idxs = train_idxs_array[train_idxs],␣
 ↪train_idxs_array[valid_idxs]
X_train = data_df["full_text_processed"].values[train_idxs]
X_valid = data_df["full_text_processed"].values[valid_idxs]
```

```
X_test = data_df["full_text_processed"].values[test_idxs]
y_train = data_df["leading_label_cat"].values[train_idxs]
y_valid = data_df["leading_label_cat"].values[valid_idxs]
y_test = data_df["leading_label_cat"].values[test_idxs]
```

[21]:
```
y_test = data_df["leading_label_cat"].values[test_idxs]
```

[21]:
```
model_path = "bert-base-cased"
tokenizer = AutoTokenizer.from_pretrained(model_path)
train_encodings = tokenizer(list(X_train), truncation=True, padding=True)
valid_encodings = tokenizer(list(X_valid), truncation=True, padding=True)
test_encodings = tokenizer(list(X_test), truncation=True, padding=True)
```

[22]:
```
class TweetDataset(torch.utils.data.Dataset):
    def __init__(self, encodings, labels):
        self.encodings = encodings
        self.labels = labels

    def __getitem__(self, idx):
        item = {key: torch.tensor(val[idx]) for key, val in self.encodings.
 ↪items()}
        item["labels"] = torch.tensor(self.labels[idx])
        return item

    def __len__(self):
        return len(self.labels)


train_dataset = TweetDataset(train_encodings, y_train)
valid_dataset = TweetDataset(valid_encodings, y_valid)
test_dataset = TweetDataset(test_encodings, y_test)
```

[23]:
```
from datasets import load_metric


metric = load_metric("accuracy")

def compute_metrics(eval_pred):
    predictions, labels = eval_pred
    predictions = np.argmax(predictions, axis=1)
    return metric.compute(predictions=predictions, references=labels)
```

```
/tmp/ipykernel_7394/1904029880.py:4: FutureWarning: load_metric is deprecated
and will be removed in the next major version of datasets. Use 'evaluate.load'
instead, from the new library  Evaluate: https://huggingface.co/docs/evaluate
  metric = load_metric("accuracy")
```

[24]:
```
from transformers import AutoModelForSequenceClassification, Trainer,␣
↪TrainingArguments


batch_size = 16
n_epoch = 3
total_steps = round(len(y_train) / batch_size * n_epoch)
warmup_steps = total_steps * 0.5

training_args = TrainingArguments(
    output_dir="/tmp2/kch/ece219/results",
    learning_rate=1e-5,
    num_train_epochs=n_epoch,                # total number of training epochs
    per_device_train_batch_size=batch_size,  # batch size per device during␣
↪training
    per_device_eval_batch_size=64,           # batch size for evaluation
    warmup_steps=warmup_steps,               # number of warmup steps for␣
↪learning rate scheduler
    weight_decay=0.01,                       # strength of weight decay
    logging_dir="./logs",                    # directory for storing logs
    logging_steps=100,
    evaluation_strategy="steps",
    eval_steps=1000,
    save_strategy="steps",
    save_steps=1000,
    metric_for_best_model="eval_loss",
    load_best_model_at_end=True,
)

model = AutoModelForSequenceClassification.from_pretrained(model_path,␣
↪num_labels=3)

trainer = Trainer(
    model=model,                         # the instantiated  Transformers␣
↪model to be trained
    args=training_args,                  # training arguments, defined above
    train_dataset=train_dataset,         # training dataset
    eval_dataset=valid_dataset,          # evaluation dataset
    compute_metrics=compute_metrics,
)

trainer.train()
```

```
Some weights of the model checkpoint at bert-base-cased were not used when
initializing BertForSequenceClassification:
['cls.predictions.transform.dense.bias', 'cls.predictions.bias',
'cls.predictions.transform.LayerNorm.bias',
```

```
'cls.predictions.transform.LayerNorm.weight', 'cls.seq_relationship.weight',
'cls.predictions.transform.dense.weight', 'cls.seq_relationship.bias',
'cls.predictions.decoder.weight']
- This IS expected if you are initializing BertForSequenceClassification from
the checkpoint of a model trained on another task or with another architecture
(e.g. initializing a BertForSequenceClassification model from a
BertForPreTraining model).
- This IS NOT expected if you are initializing BertForSequenceClassification
from the checkpoint of a model that you expect to be exactly identical
(initializing a BertForSequenceClassification model from a
BertForSequenceClassification model).
Some weights of BertForSequenceClassification were not initialized from the
model checkpoint at bert-base-cased and are newly initialized:
['classifier.weight', 'classifier.bias']
You should probably TRAIN this model on a down-stream task to be able to use it
for predictions and inference.
/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/transformers/optimization.py:306: FutureWarning: This implementation of
AdamW is deprecated and will be removed in a future version. Use the PyTorch
implementation torch.optim.AdamW instead, or set `no_deprecation_warning=True`
to disable this warning
  warnings.warn(
***** Running training *****
  Num examples = 40158
  Num Epochs = 3
  Instantaneous batch size per device = 16
  Total train batch size (w. parallel, distributed & accumulation) = 16
  Gradient Accumulation steps = 1
  Total optimization steps = 7530
  Number of trainable parameters = 108312579

<IPython.core.display.HTML object>

***** Running Evaluation *****
  Num examples = 10040
  Batch size = 64
Saving model checkpoint to /tmp2/kch/ece219/results/checkpoint-1000
Configuration saved in /tmp2/kch/ece219/results/checkpoint-1000/config.json
Model weights saved in
/tmp2/kch/ece219/results/checkpoint-1000/pytorch_model.bin
***** Running Evaluation *****
  Num examples = 10040
  Batch size = 64
Saving model checkpoint to /tmp2/kch/ece219/results/checkpoint-2000
Configuration saved in /tmp2/kch/ece219/results/checkpoint-2000/config.json
Model weights saved in
/tmp2/kch/ece219/results/checkpoint-2000/pytorch_model.bin
***** Running Evaluation *****
  Num examples = 10040
```

```
  Batch size = 64
Saving model checkpoint to /tmp2/kch/ece219/results/checkpoint-3000
Configuration saved in /tmp2/kch/ece219/results/checkpoint-3000/config.json
Model weights saved in
/tmp2/kch/ece219/results/checkpoint-3000/pytorch_model.bin
***** Running Evaluation *****
  Num examples = 10040
  Batch size = 64
Saving model checkpoint to /tmp2/kch/ece219/results/checkpoint-4000
Configuration saved in /tmp2/kch/ece219/results/checkpoint-4000/config.json
Model weights saved in
/tmp2/kch/ece219/results/checkpoint-4000/pytorch_model.bin
***** Running Evaluation *****
  Num examples = 10040
  Batch size = 64
Saving model checkpoint to /tmp2/kch/ece219/results/checkpoint-5000
Configuration saved in /tmp2/kch/ece219/results/checkpoint-5000/config.json
Model weights saved in
/tmp2/kch/ece219/results/checkpoint-5000/pytorch_model.bin
***** Running Evaluation *****
  Num examples = 10040
  Batch size = 64
Saving model checkpoint to /tmp2/kch/ece219/results/checkpoint-6000
Configuration saved in /tmp2/kch/ece219/results/checkpoint-6000/config.json
Model weights saved in
/tmp2/kch/ece219/results/checkpoint-6000/pytorch_model.bin
***** Running Evaluation *****
  Num examples = 10040
  Batch size = 64
Saving model checkpoint to /tmp2/kch/ece219/results/checkpoint-7000
Configuration saved in /tmp2/kch/ece219/results/checkpoint-7000/config.json
Model weights saved in
/tmp2/kch/ece219/results/checkpoint-7000/pytorch_model.bin


Training completed. Do not forget to share your model on huggingface.co/models
=)


Loading best model from /tmp2/kch/ece219/results/checkpoint-6000 (score:
0.7854036688804626).
```

[24]:
```
TrainOutput(global_step=7530, training_loss=0.8443113258635381,
metrics={'train_runtime': 6172.508, 'train_samples_per_second': 19.518,
'train_steps_per_second': 1.22, 'total_flos': 2.3959476793768572e+16,
'train_loss': 0.8443113258635381, 'epoch': 3.0})
```

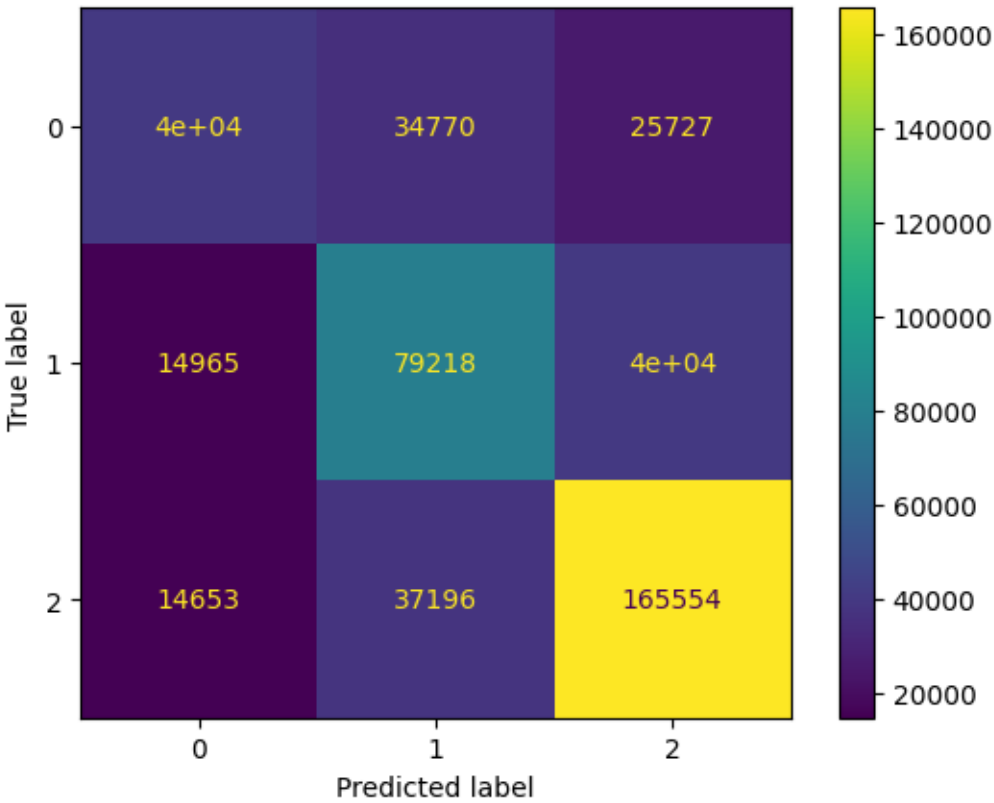
[25]:
```
y_pred = trainer.predict(test_dataset).predictions.argmax(axis=1)
```

```
***** Running Prediction *****
  Num examples = 451778
  Batch size = 64
```

```
<IPython.core.display.HTML object>
```

[26]:
```
evaluate_mul(y_test, y_pred, average=None, clf=None)
```

```
====================
```



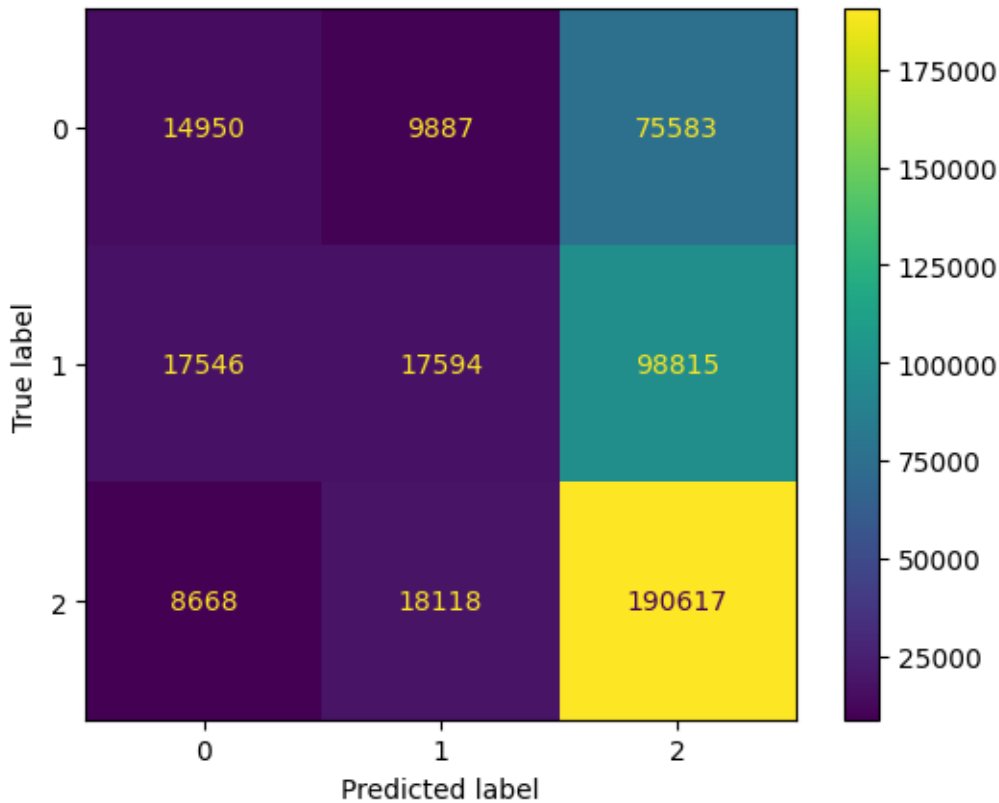

```
Accuracy: 0.63017
Recall: [0.39756025 0.5913777  0.76150743]
Precision: [0.57409298 0.52398402 0.71651959]
F-1 score: [0.46979013 0.55564479 0.73832884]
====================
```

[26]: 0.63017

0.0.3 Weak Baselines

[30]:
```
from collections import Counter


def rule_based_predict(text):
    lower_text = text.lower()
    sea_kws = [
        "seahawk", "seattle", "marshawn", "lynch", "steven", "hauschka",
        "chris", "matthews", "russell", "wilson", "doug", "baldwin",
    ]
    ne_kws = [
        "patriot", "newengland", "tom", "brady", "stephen", "gostkowski",
        "brandon", "lafell", "rob", "gronkowski", "danny", "amendola",
        "julian", "edelman",
    ]
    sea = ne = False

    for sea_kw in sea_kws:
        if sea_kw in lower_text:
            sea = True
            break

    for ne_kw in ne_kws:
        if ne_kw in lower_text:
            ne = True
            break

    if ne and not sea:
        return 0
    if sea and not ne:
        return 1
    return 2


y_pred_rule_based = [rule_based_predict(t) for t in tqdm(test_mul["full_text"])]
print(Counter(y_pred_rule_based))
evaluate_mul(y_test, y_pred_rule_based, average=None, clf=None)
```

```
  0%|          | 0/451778 [00:00<?, ?it/s]

Counter({2: 365015, 1: 45599, 0: 41164})
====================
```




```
Accuracy: 0.49396
Recall: [0.14887473 0.13134262 0.87679103]
Precision: [0.36318142 0.38584179 0.5222169 ]
F-1 score: [0.21118205 0.19597447 0.65457112]
====================
```

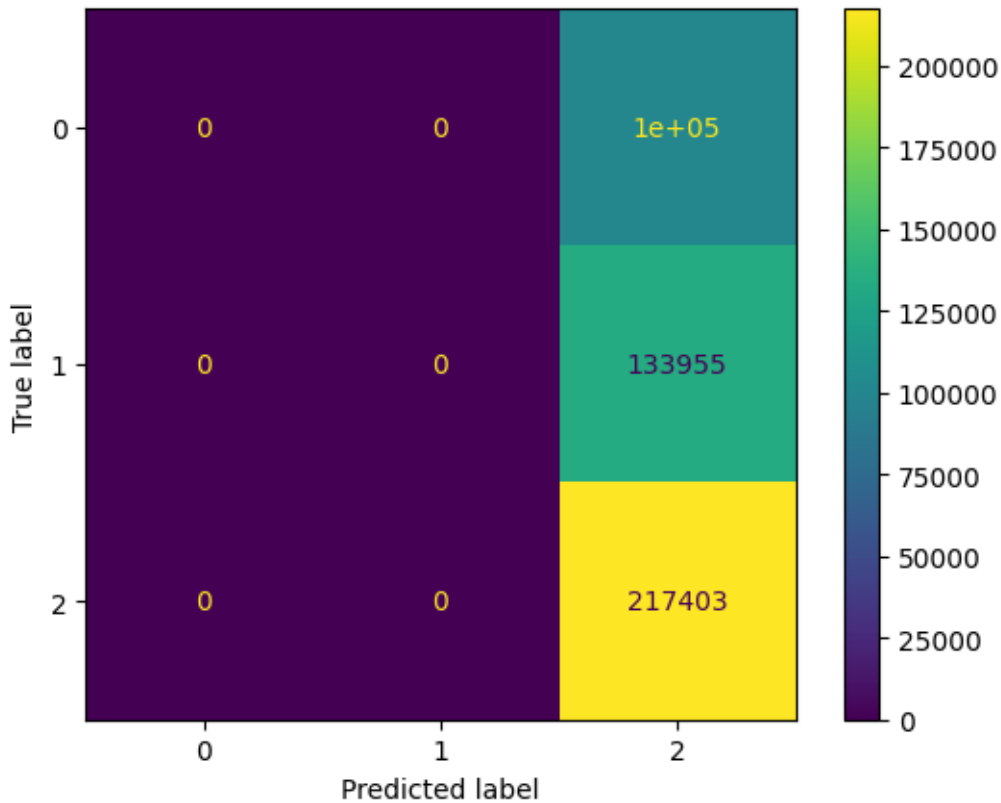
[30]: 0.49396

[28]:
```
round(Counter(test_mul["leading_label"])["TIE"] / len(test_idxs), 5)
```

[28]: 0.48122

[25]:
```
evaluate_mul(y_test, [2] * len(y_test), average=None, clf=None)
```

```
====================
```




```
Accuracy: 0.48122
Recall: [0. 0. 1.]

/home/ray080186/.pyenv/versions/3.9.0/lib/python3.9/site-
packages/sklearn/metrics/_classification.py:1344: UndefinedMetricWarning:
Precision is ill-defined and being set to 0.0 in labels with no predicted
samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))

Precision: [0.         0.         0.48121644]
F-1 score: [0.         0.         0.64975844]
====================
```

[25]: 0.48122

[29]:
```
### Data
# normalize (o)
# normalize advanced (o)
# hashtag split (o)
# sentiment (o)
# emoji (o)
```

```
### Model
# w NMF (o)
# w SVD (o)
# ensemble
# fasttext
# roberta or xxbert

### Baseline
# 2+
# raw data
# no extra preprocess

### Report


############ TODO #############

# 1. Pipeline [done]
# 2. Baseline
# 3. textrnngen
# 4. Report
```

P4_Q10_Textgenrnn

March 9, 2023

[1]:
```
from keras.utils import pad_sequences
```

[14]:
```
from datetime import datetime
import tensorflow as tf
# import tensorflow.compat.v1 as tf
# tf.disable_v2_behavior()

import numpy as np
import os
import time

%load_ext autoreload
%autoreload 2

from keras.utils import plot_model
from textgenrnn import textgenrnn
import matplotlib as mpl
import matplotlib.pyplot as plt
%matplotlib inline
```

```
The autoreload extension is already loaded. To reload it, use:
  %reload_ext autoreload
```

[15]:
```
# test_cuda = tf.constant("test cuda")
# with tf.Session() as sess:
#     print(tf.test.is_gpu_available(cuda_only=True))
#     print(sess.run(test_cuda))
```

[16]:
```
mpl.rcParams[u'figure.figsize'] = [12, 8]
mpl.rcParams[u'figure.dpi'] = 300
```

[33]:
```
import matplotlib.pylab as pylab
params = {'legend.fontsize': 'x-large',
          'figure.figsize': (15, 5),
         'axes.labelsize': 'x-large',
         'axes.titlesize':'x-large',
         'xtick.labelsize':'x-large',
         'ytick.labelsize':'x-large'}
```

```python
pylab.rcParams.update(params)

fsize = 15
tsize = 18
tdir = 'in'
major = 5.0
minor = 3.0
lwidth = 0.8
lhandle = 2.0
plt.style.use('default')
plt.rcParams['text.usetex'] = True
plt.rcParams['font.size'] = fsize
plt.rcParams['legend.fontsize'] = tsize
plt.rcParams['xtick.direction'] = tdir
plt.rcParams['ytick.direction'] = tdir
plt.rcParams['xtick.major.size'] = major
plt.rcParams['xtick.minor.size'] = minor
plt.rcParams['ytick.major.size'] = 5.0
plt.rcParams['ytick.minor.size'] = 3.0
plt.rcParams['axes.linewidth'] = lwidth
plt.rcParams['legend.handlelength'] = lhandle

colors = ['#0051a2', '#97964a', '#ffd44f', '#f4777f', '#93003a']


def plot_result(history):
    plt.figure(figsize=(16,6))
    plt.subplot(1,2,1)
    plt.plot(history.history['loss'], color=colors[0])
    plt.plot(history.history['val_loss'], color=colors[-1])
#     plt.title('Loss')
    plt.ylabel('Loss')
    plt.xlabel('Epoch')
    plt.legend(['Train', 'Validation'])
    plt.grid(alpha = .2)

    plt.subplot(1,2,2)
    # print(history.history)
    plt.plot(history.history['accuracy'], color=colors[0])
    plt.plot(history.history['val_accuracy'], color=colors[-1])
#     plt.title('Accuracy')
    plt.ylabel('Accuracy')
    plt.xlabel('Epoch')
    plt.legend(['Train', 'Validation'])
    plt.grid(alpha = .2)

    plt.show()
```

1 Baseline Setting

1.1 3-layer Basic LSTM network

[5]:
```
model_cfg = {
    'word_level': True,   # set to True if want to train a word-level model␣
 ↪(requires more data and smaller max_length)
    'rnn_size': 256,   # number of LSTM cells of each layer (128/256␣
 ↪recommended)
    'rnn_layers': 3,   # number of LSTM layers (>=2 recommended)
    'rnn_bidirectional': True,   # consider text both forwards and backward,␣
 ↪can give a training boost
    'max_length': 15,   # number of tokens to consider before predicting the␣
 ↪next (20-40 for characters, 5-10 for words recommended)
    'max_words': 10000,   # maximum number of words to model; the rest will be␣
 ↪ignored (word-level model only)
    'use_attention': False
}

train_cfg = {
    'line_delimited': True,   # set to True if each text has its own line in␣
 ↪the source file
    'num_epochs': 10,   # set higher to train the model for longer
    'gen_epochs': 5,   # generates sample text from model after given number of␣
 ↪epochs
    'train_size': 0.8,   # proportion of input data to train on: setting < 1.0␣
 ↪limits model from learning perfectly
    'dropout': 0.0,   # ignore a random proportion of source tokens each epoch,␣
 ↪allowing model to generalize better
    'validation': True,   # If train__size < 1.0, test on holdout dataset; will␣
 ↪make overall training slower
    'is_csv': False   # set to True if file is a CSV exported from Excel/
 ↪BigQuery/pandas
}
```

[6]:
```
gopatriots_textgen = textgenrnn(config=model_cfg, name='gopatriots',␣
 ↪new_model=True)
# plot_model(textgen.model, show_shapes=True)
```

```
Metal device set to: Apple M2

2023-03-07 12:07:33.071432: I
tensorflow/core/common_runtime/pluggable_device/pluggable_device_factory.cc:306]
Could not identify NUMA node of platform GPU ID 0, defaulting to 0. Your kernel
may not have been built with NUMA support.
2023-03-07 12:07:33.071450: I
tensorflow/core/common_runtime/pluggable_device/pluggable_device_factory.cc:272]
```

```
Created TensorFlow device (/job:localhost/replica:0/task:0/device:GPU:0 with 0
MB memory) -> physical PluggableDevice (device: 0, name: METAL, pci bus id:
<undefined>)

Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input (InputLayer)          [(None, 15)]              0

 embedding (Embedding)       (None, 15, 100)           46500

 rnn_1 (Bidirectional)       (None, 15, 512)           733184

 rnn_2 (Bidirectional)       (None, 15, 512)           1576960

 rnn_3 (Bidirectional)       (None, 512)               1576960

 output (Dense)              (None, 465)               238545

=================================================================
Total params: 4,172,149
Trainable params: 4,172,149
Non-trainable params: 0
_________________________________________________________________
```

[7]:
```
gopatriots_history = gopatriots_textgen.train_from_file(
    file_path="../gopatriots.txt",
    new_model=True,
    num_epochs=train_cfg['num_epochs'],
    gen_epochs=train_cfg['gen_epochs'],
    batch_size=512,
    train_size=train_cfg['train_size'],
    dropout=train_cfg['dropout'],
    validation=train_cfg['validation'],
    is_csv=train_cfg['is_csv'],
    rnn_layers=model_cfg['rnn_layers'],
    rnn_size=model_cfg['rnn_size'],
    rnn_bidirectional=model_cfg['rnn_bidirectional'],
    max_length=model_cfg['max_length'],
    dim_embeddings=100,
    word_level=model_cfg['word_level']
)
```

```
16,994 texts collected.
Training new model w/ 3-layer, 256-cell Bidirectional LSTMs
Model: "model_1"
_________________________________________________________________
```

```
 Layer (type)                Output Shape              Param #
=================================================================
 input (InputLayer)          [(None, 15)]              0

 embedding (Embedding)       (None, 15, 100)           1000200

 rnn_1 (Bidirectional)       (None, 15, 512)           733184

 rnn_2 (Bidirectional)       (None, 15, 512)           1576960

 rnn_3 (Bidirectional)       (None, 512)               1576960

 output (Dense)              (None, 10002)             5131026

=================================================================
Total params: 10,018,330
Trainable params: 10,018,330
Non-trainable params: 0
_________________________________________________________________
Training on 193,790 word sequences.
Epoch 1/10

/Users/sinyapeng/Desktop/textgenrnn/textgenrnn/textgenrnn.py:210: UserWarning:
`Model.fit_generator` is deprecated and will be removed in a future version.
Please use `Model.fit`, which supports generators.
  history = model_t.fit_generator(gen, steps_per_epoch=steps_per_epoch,
2023-03-07 12:07:35.253351: W
tensorflow/tsl/platform/profile_utils/cpu_utils.cc:128] Failed to get CPU
frequency: 0 Hz
2023-03-07 12:07:36.770370: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

378/378 [==============================] - ETA: 0s - loss: 5.9899 - accuracy:
0.0703

2023-03-07 12:08:48.602653: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

378/378 [==============================] - 81s 208ms/step - loss: 5.9899 -
accuracy: 0.0703 - val_loss: 5.6364 - val_accuracy: 0.0741 - lr: 0.0040
Epoch 2/10
378/378 [==============================] - 78s 205ms/step - loss: 5.0582 -
accuracy: 0.1748 - val_loss: 4.6783 - val_accuracy: 0.2679 - lr: 0.0036
Epoch 3/10
378/378 [==============================] - 85s 224ms/step - loss: 4.3349 -
accuracy: 0.3114 - val_loss: 4.3380 - val_accuracy: 0.3450 - lr: 0.0032
Epoch 4/10
378/378 [==============================] - 94s 248ms/step - loss: 4.0397 -
```

```
accuracy: 0.3505 - val_loss: 4.2174 - val_accuracy: 0.3594 - lr: 0.0028
Epoch 5/10
378/378 [==============================] - ETA: 0s - loss: 3.8462 - accuracy:
0.3662####################
Temperature: 0.2
####################

2023-03-07 12:14:45.607774: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

1/1 [==============================] - 1s 938ms/step
1/1 [==============================] - 0s 34ms/step
1/1 [==============================] - 0s 34ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 34ms/step
1/1 [==============================] - 0s 34ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 33ms/step
1/1 [==============================] - 0s 33ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
" i ' m not a great game . . . . . #gopatriots

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 32ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
```

```
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
" i ' m not a great game ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !

1/1 [==============================] - 0s 23ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
" " " " " " " " " i ' m a great game ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !

####################
Temperature: 0.5
####################
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
" " " " " " champions ! ! #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
" patriots for know i ' m ready for the lucky ! ! ! #gopatriots

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
" i love it ! #gopatriots

####################
Temperature: 1.0
####################
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
super bowl ta minutes sucks will talk ! ! ! ! #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
" weak  i did that will o super bowl this luck in like they ' m . #gopatriots
#patriotswin

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
only jv

378/378 [==============================] - 119s 314ms/step - loss: 3.8462 -
accuracy: 0.3662 - val_loss: 4.1894 - val_accuracy: 0.3636 - lr: 0.0024
Epoch 6/10
378/378 [==============================] - 90s 239ms/step - loss: 3.7091 -
accuracy: 0.3749 - val_loss: 4.2199 - val_accuracy: 0.3681 - lr: 0.0020
Epoch 7/10
378/378 [==============================] - 94s 249ms/step - loss: 3.5954 -
accuracy: 0.3821 - val_loss: 4.1911 - val_accuracy: 0.3737 - lr: 0.0016
Epoch 8/10
378/378 [==============================] - 94s 250ms/step - loss: 3.4708 -
accuracy: 0.3902 - val_loss: 4.2218 - val_accuracy: 0.3769 - lr: 0.0012
Epoch 9/10
378/378 [==============================] - 94s 250ms/step - loss: 3.3762 -
accuracy: 0.3951 - val_loss: 4.2122 - val_accuracy: 0.3807 - lr: 8.0000e-04
Epoch 10/10
378/378 [==============================] - ETA: 0s - loss: 3.2930 - accuracy:
0.4011####################
Temperature: 0.2
####################
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
" " " : patriots

1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
#gopatriots

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
" " " : #gopatriots

####################
Temperature: 0.5
####################
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
" " " #gopatriots

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
" " " : http : / / t . co /

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
#gopatriots "

####################
Temperature: 1.0
####################
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 925s 925s/step
1/1 [==============================] - 0s 38ms/step
1/1 [==============================] - 0s 36ms/step
1/1 [==============================] - 0s 23ms/step
" ) #superbowlxlix brady are weird !

1/1 [==============================] - 0s 23ms/step
```

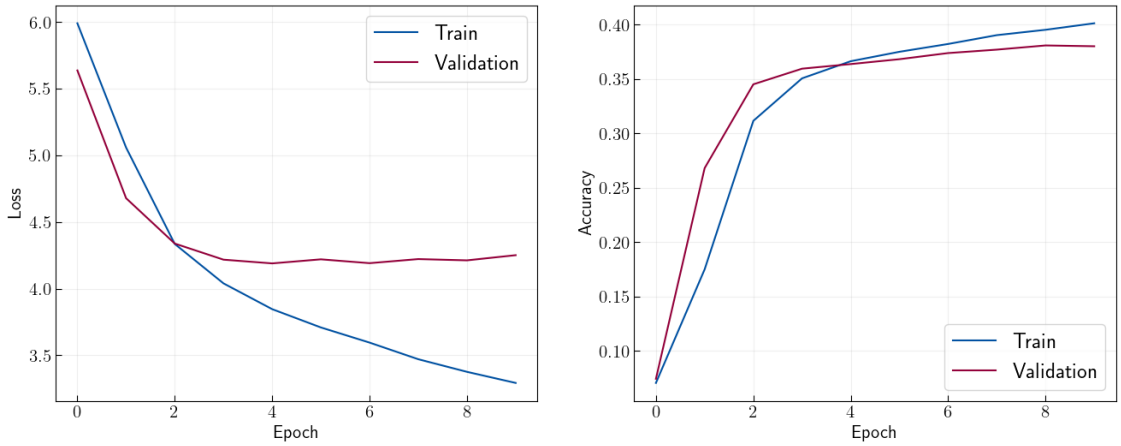
```
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
http : / / t . co /

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
atta from let ' s get #halftime ? ? ?  #gopatriots

378/378 [==============================] - 2854s 8s/step - loss: 3.2930 -
accuracy: 0.4011 - val_loss: 4.2510 - val_accuracy: 0.3800 - lr: 4.0000e-04
```

[8]:
```
gopatriots_textgen.model.summary()
```

```
Model: "model_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input (InputLayer)          [(None, 15)]              0

 embedding (Embedding)       (None, 15, 100)           1000200

 rnn_1 (Bidirectional)       (None, 15, 512)           733184

 rnn_2 (Bidirectional)       (None, 15, 512)           1576960

 rnn_3 (Bidirectional)       (None, 512)               1576960

 output (Dense)              (None, 10002)             5131026

=================================================================
Total params: 10,018,330
```

```
Trainable params: 10,018,330
Non-trainable params: 0
_________________________________________________________________
```

[34]:
```
plot_result(history)
```



[10]:
```
gopatriots_textgen.generate(20, temperature=1.0)
```

```
  0%|
| 0/20 [00:00<?, ?it/s]

1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step

  5%|
| 1/20 [00:00<00:07,  2.49it/s]

7 http : / / t . co / "

1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step

 10%|
| 2/20 [00:00<00:06,  2.73it/s]

pats ! ! ! ! ! ! ! #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step

 15%|
| 3/20 [00:00<00:05,  3.26it/s]

great job nope . #superbowi #gopatriots

1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 40ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

 20%|
| 4/20 [00:01<00:05,  3.16it/s]

@katyperry i like you guys gear ? #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step

 25%|
| 5/20 [00:01<00:04,  3.20it/s]
```

```
this game girls fans games . #superbowlxlix #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step

 30%|
| 6/20 [00:02<00:05,  2.39it/s]

tom brady  ! we go to buy yourself ! about deflated nights who left .
#gopatriots "

1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step

 35%|
| 7/20 [00:02<00:05,  2.41it/s]

nice to finish go with this ? ! #ontoseattle #pepsisbhalftime #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step

 40%|
| 8/20 [00:03<00:06,  1.96it/s]

" a #superbowl happy , in colts the bets winning now family during the game
begin ! #gopatriots ! !

1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step

 45%|
| 9/20 [00:04<00:09,  1.20it/s]

" este ending to hate out of a #freebook http : / / t . co / azo8xkkkf2 enjoy
you all you - ? ! love ? ! #gopatriots | #uphoenixstadium ll have a la feelin
@colts #td #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step

 50%|
| 10/20 [00:05<00:07,  1.35it/s]

" retweet early to see 10 seconds a good superbowl ! ! js #gopatriots

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 21ms/step

 55%|
| 11/20 [00:05<00:05,  1.67it/s]

that game time so cute ! #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

 60%|
| 12/20 [00:05<00:04,  2.00it/s]

#superbowl #superbowl td #cheers #sb49 #gopatriots "

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

 65%|
| 13/20 [00:07<00:04,  1.46it/s]

thank u points 4 : #luck today . football history ! ! i wanna get gronkowski .
and burn years to play ! http : / / t . co /

1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step

 70%|
| 14/20 [00:07<00:03,  1.89it/s]

" #patriots #gopatriots "

1/1 [==============================] - 0s 21ms/step


1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

 80%|
                          | 16/20 [00:07<00:01,  2.58it/s]

! #gopatriots #superbowl http : / / t . co /

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

 85%|
                        | 17/20 [00:07<00:00,  3.16it/s]

#patsnation

1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step

 90%|
                      | 18/20 [00:08<00:00,  3.01it/s]

someone hurt i need to use the pats so #gopatriots

1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

 95%|
                    | 19/20 [00:08<00:00,  3.70it/s]

yaaas #gopatriots

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

100%|
                  | 20/20 [00:08<00:00,  2.33it/s]

1 ma a ass ! #gopatriots "
```

1.1.1 Clean up the training set to exclude the overwhelming "I will be interviewed ..."

[22]:
```
model_cfg = {
    'word_level': True,   # set to True if want to train a word-level model␣
↪(requires more data and smaller max_length)
    'rnn_size': 128,   # number of LSTM cells of each layer (128/256␣
↪recommended)
    'rnn_layers': 3,   # number of LSTM layers (>=2 recommended)
```

```
    'rnn_bidirectional': True,   # consider text both forwards and backward,␣
↪can give a training boost
    'max_length': 15,   # number of tokens to consider before predicting the␣
↪next (20-40 for characters, 5-10 for words recommended)
    'max_words': 10000,   # maximum number of words to model; the rest will be␣
↪ignored (word-level model only)
    'use_attention': False
}

train_cfg = {
    'line_delimited': True,   # set to True if each text has its own line in␣
↪the source file
    'num_epochs': 6,   # set higher to train the model for longer
    'gen_epochs': 5,   # generates sample text from model after given number of␣
↪epochs
    'train_size': 0.8,   # proportion of input data to train on: setting < 1.0␣
↪limits model from learning perfectly
    'dropout': 0.0,   # ignore a random proportion of source tokens each epoch,␣
↪allowing model to generalize better
    'validation': True,   # If train__size < 1.0, test on holdout dataset; will␣
↪make overall training slower
    'is_csv': False   # set to True if file is a CSV exported from Excel/
↪BigQuery/pandas
}
```

[23]:
```
gohawks_textgen = textgenrnn(config=model_cfg, name='gohawks', new_model=True)
gohawks_textgen.model.summary()

start_time = time.time()

gohawks_history = gohawks_textgen.train_from_file(
    file_path="../gohawks.txt",
    new_model=True,
    num_epochs=train_cfg['num_epochs'],
    gen_epochs=train_cfg['gen_epochs'],
    batch_size=512,
    train_size=train_cfg['train_size'],
    dropout=train_cfg['dropout'],
    validation=train_cfg['validation'],
    is_csv=train_cfg['is_csv'],
    rnn_layers=model_cfg['rnn_layers'],
    rnn_size=model_cfg['rnn_size'],
    rnn_bidirectional=model_cfg['rnn_bidirectional'],
    max_length=model_cfg['max_length'],
    dim_embeddings=100,
    word_level=model_cfg['word_level']
```

```
)
elapsed_time = time.time() - start_time
print("Time spent: ", elapsed_time, " seconds")
```

```
Model: "model_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input (InputLayer)          [(None, 15)]              0

 embedding (Embedding)       (None, 15, 100)           46500

 rnn_1 (Bidirectional)       (None, 15, 256)           235520

 rnn_2 (Bidirectional)       (None, 15, 256)           395264

 rnn_3 (Bidirectional)       (None, 256)               395264

 output (Dense)              (None, 465)               119505

=================================================================
Total params: 1,192,053
Trainable params: 1,192,053
Non-trainable params: 0
_________________________________________________________________
Model: "model_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input (InputLayer)          [(None, 15)]              0

 embedding (Embedding)       (None, 15, 100)           46500

 rnn_1 (Bidirectional)       (None, 15, 256)           235520

 rnn_2 (Bidirectional)       (None, 15, 256)           395264

 rnn_3 (Bidirectional)       (None, 256)               395264

 output (Dense)              (None, 465)               119505

=================================================================
Total params: 1,192,053
Trainable params: 1,192,053
Non-trainable params: 0
_________________________________________________________________
158,604 texts collected.
```

```
Training new model w/ 3-layer, 128-cell Bidirectional LSTMs
Model: "model_5"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input (InputLayer)          [(None, 15)]              0

 embedding (Embedding)       (None, 15, 100)           1000200

 rnn_1 (Bidirectional)       (None, 15, 256)           235520

 rnn_2 (Bidirectional)       (None, 15, 256)           395264

 rnn_3 (Bidirectional)       (None, 256)               395264

 output (Dense)              (None, 10002)             2570514

=================================================================
Total params: 4,596,762
Trainable params: 4,596,762
Non-trainable params: 0
_________________________________________________________________
Training on 2,242,232 word sequences.

/Users/sinyapeng/Desktop/textgenrnn/textgenrnn/textgenrnn.py:210: UserWarning:
`Model.fit_generator` is deprecated and will be removed in a future version.
Please use `Model.fit`, which supports generators.
  history = model_t.fit_generator(gen, steps_per_epoch=steps_per_epoch,

Epoch 1/6

2023-03-07 14:14:05.244300: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

4379/4379 [==============================] - ETA: 0s - loss: 3.9501 - accuracy:
0.3632

2023-03-07 14:22:54.754059: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

4379/4379 [==============================] - 592s 134ms/step - loss: 3.9501 -
accuracy: 0.3632 - val_loss: 3.6665 - val_accuracy: 0.3983 - lr: 0.0040
Epoch 2/6
4379/4379 [==============================] - 591s 135ms/step - loss: 3.5490 -
accuracy: 0.4104 - val_loss: 3.5412 - val_accuracy: 0.4152 - lr: 0.0033
Epoch 3/6
4379/4379 [==============================] - 636s 145ms/step - loss: 3.4171 -
accuracy: 0.4244 - val_loss: 3.4754 - val_accuracy: 0.4233 - lr: 0.0027
Epoch 4/6
```

```
4379/4379 [==============================] - 597s 136ms/step - loss: 3.3267 -
accuracy: 0.4337 - val_loss: 3.4366 - val_accuracy: 0.4299 - lr: 0.0020
Epoch 5/6
4379/4379 [==============================] - ETA: 0s - loss: 3.2444 - accuracy:
0.4424####################
Temperature: 0.2
####################

2023-03-07 15:04:29.226000: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

1/1 [==============================] - 2s 2s/step
1/1 [==============================] - 0s 77ms/step
1/1 [==============================] - 0s 72ms/step
1/1 [==============================] - 0s 73ms/step
1/1 [==============================] - 0s 71ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 66ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 45ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 75ms/step
1/1 [==============================] - 0s 22ms/step
" " " : i ' m a little bit of the seahawks . #gohawks " " "

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 23ms/step
```

```
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 29ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 29ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 23ms/step
```

```
1/1 [==============================] - 0s 30ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
" " " : i ' m a little bit of the game . . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
```

```
. . . . . .
. . . . . .
. . . . . .

1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
" i ' m not a seahawks fan , but i ' m a little and the seahawks are going to
the super bowl ! ! ! #gohawks

####################
Temperature: 0.5
####################
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 19ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
what a play ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
" " " @lanadelcunt : i hope the seahawks beat seattle ð  ð  ð   #gohawks " "
"

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 37ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 42ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 25ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
```

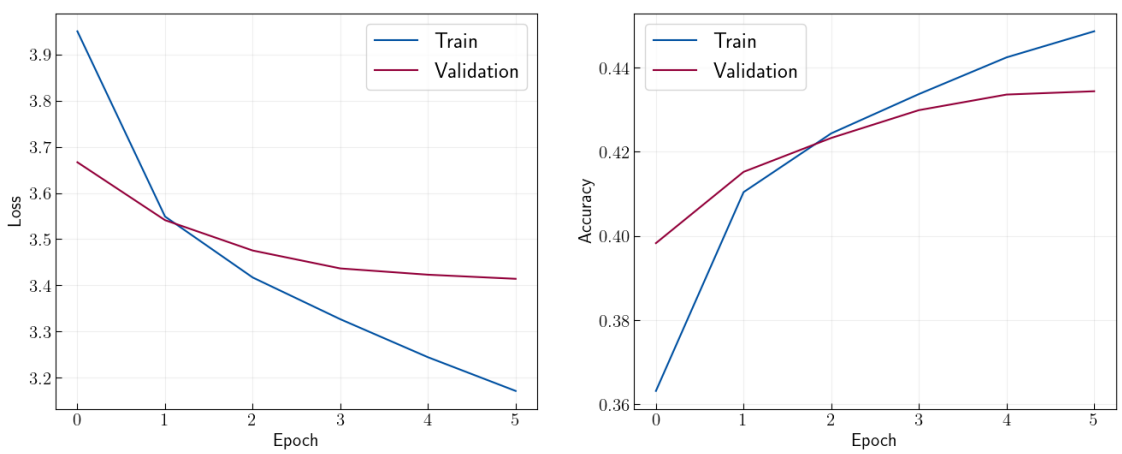
```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
" " " : " " : we are going to the super bowl ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
```

```
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !

####################
Temperature: 1.0
####################
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
#superbowlxlix #superbowlxlix #gohawks #seahawksvspackers #nfcchampionship "

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
aaron rogers better much balls having to come . there started to get to the next
5 years . . let ' s get strong ! #gohawks #12s #louder #weare12
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
im ' em tomorrow ! ! #sbmediaday #gohawks #marshawnlynch #sbmediaday
#patriotsnation

4379/4379 [==============================] - 646s 147ms/step - loss: 3.2444 -
accuracy: 0.4424 - val_loss: 3.4231 - val_accuracy: 0.4336 - lr: 0.0013
Epoch 6/6
4379/4379 [==============================] - 584s 133ms/step - loss: 3.1714 -
accuracy: 0.4486 - val_loss: 3.4142 - val_accuracy: 0.4344 - lr: 6.6667e-04
Time spent:  3760.1543078422546  seconds
```

[35]:
```
plot_result(gohawks_history)
```



[26]:
```
gohawks_textgen.generate(20, temperature=1.0)
```

```
  0%|
| 0/20 [00:00<?, ?it/s]

1/1 [==============================] - 0s 31ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step

  5%|
| 1/20 [00:00<00:03,  5.16it/s]

cmon hawks ! #gohawks

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step

 10%|
| 2/20 [00:00<00:04,  4.02it/s]

the man must have happened ! #gohawks

1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

 15%|
| 3/20 [00:01<00:07,  2.35it/s]

@seattlehawks _ 12 it ' s going to be positive . in the playoffs #sb49 #gohawks

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
```

```
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step

 20%|
| 4/20 [00:01<00:09,  1.72it/s]

" hey rest , i support football music and woke up for the game . well played
nothing . #gohawks "

1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step

 25%|
| 5/20 [00:02<00:09,  1.62it/s]

" from up at home today , it ' s been crazy was there .   #gohawks "

1/1 [==============================] - 0s 30ms/step
1/1 [==============================] - 0s 25ms/step
```

```
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 42ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step

 30%|
| 6/20 [00:03<00:08,  1.57it/s]

1 - 0 seattle over boston & amp ; be safe and got this backs #gohawks

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step

 35%|
| 7/20 [00:03<00:07,  1.78it/s]

warming up by @dangerusswilson this is how he loves you #gohawks

1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 31ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 21ms/step

 40%|
| 8/20 [00:04<00:07,  1.50it/s]

: brady is member @seahawks #gohawks #bluefriday #sb49 @ seatac in the coffee :
http : / / t . co /

1/1 [==============================] - 0s 22ms/step


1/1 [==============================] - 0s 22ms/step


1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 28ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 28ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 35ms/step
1/1 [==============================] - 0s 37ms/step
1/1 [==============================] - 0s 30ms/step
1/1 [==============================] - 0s 37ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step

 55%|
| 11/20 [00:05<00:04,  2.12it/s]

i don ' t mind may hold sick . haha #gohawks http : / / t . co /
```

```
1/1 [==============================] - 0s 29ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 28ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 29ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 25ms/step

 60%|
| 12/20 [00:06<00:03,  2.09it/s]

" @chris _ aww guys , #gohawks #12thman , "

1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 28ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 32ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 25ms/step

 65%|
| 13/20 [00:06<00:03,  1.80it/s]

wonder game busted time to make day pre - game raising the chips . #gbvssea
#gohawks bee say

1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
```

```
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step

 70%|
| 14/20 [00:07<00:03,  1.52it/s]

" @dougbaldwinjr in russ always can ' t " " pedestrian " " . we ' re on fb cliff
#13 #gohawks "

1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step

 75%|
                              | 15/20 [00:08<00:03,  1.44it/s]

everybody knows the pats never gave up as omg #gohawks ( video ) . stupid bombs
```

```
like enough .

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step

 80%|
                          | 16/20 [00:09<00:03,  1.31it/s]

" i was jacked about a td , will you mean . can ' t wait till next sunday .
#gohawks ! "

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 21ms/step
```

```
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 23ms/step

 85%|
                        | 17/20 [00:10<00:02,  1.36it/s]

" thank you pete carroll , but he ' s gonna be together . #gohawks #sb49 "

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 23ms/step

 90%|
                      | 18/20 [00:10<00:01,  1.66it/s]

game on ! ! ! #gohawks

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 21ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step

 95%|
                    | 19/20 [00:11<00:00,  1.66it/s]

hmmm . . . . . #gohawks http : / / t . co /

1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step
1/1 [==============================] - 0s 23ms/step
1/1 [==============================] - 0s 22ms/step

100%|
```

```
                    | 20/20 [00:11<00:00,  1.77it/s]
game time #gohawks #seavsgb
```

[ ]:

```

```

[ ]:

```

```

[ ]:

```

```