Seattle Car Accident Severity

Introduction/ Business Problem

Traffic accidents cause fatalities and economic losses worldwide every year. Therefore, from the perspective of loss prevention, this is one of the main areas of social concern. According to the preliminary estimates released from the National Safety Council, approximately 38,800 people lost their lives to car crashes in 2019. Around 4.4 million people were injured seriously enough to require medical attention in crashes in 2019. Although motor vehicle fatality and casualty has declined for the second consecutive year, the number of deaths is still unacceptable. Today, the majority of new vehicle models from manufacturers includes many advanced driver assistance and safety technology, such as automatic emergency braking, lane departure warning systems, driver monitoring system and so on, all of which are proven to be effective on reducing the severity of crashes or accidents. However, it is still necessary to study characteristics of the accident. Therefore, establishing an accident severity prediction model and improving the model are the keys to improve the safety performance of the road traffic system.

Data Understanding

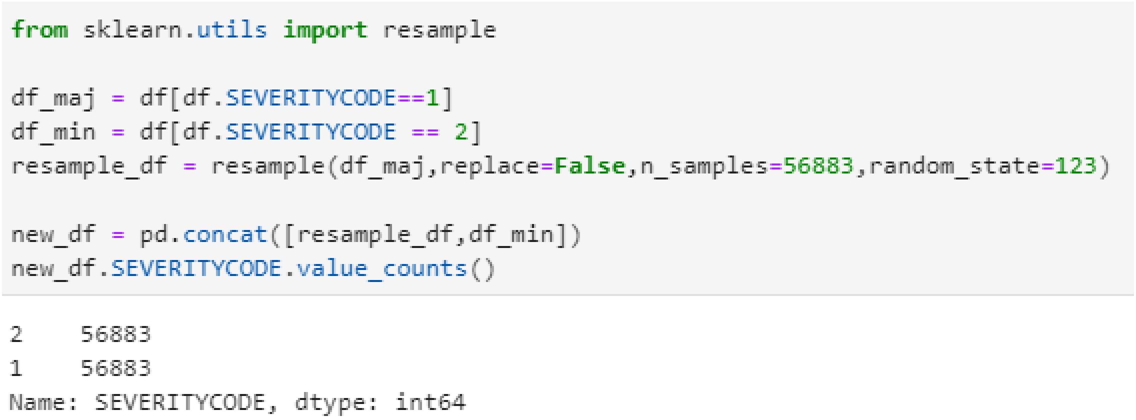
In the accident severity modeling, the input vector is the characteristics of the accident, such as the attributes of driver behavior like speeding, and environmental characteristics like weather and road condition, and the output vector is the category corresponding to the severity of the accident.

A comprehensive dataset of 194,673 accidents occurring between 2004 to 2020 in Seattle was obtained for this analysis. the dataset consists of 39 columns describing the details of each accidents. The dependent variable, “SEVERITYCODE”, contains numbers that correspond to different severity level caused by the accident. “1” indicates property damage only collision, and “2” indicates injury collision. The existing data is not ready for data analysis; therefore, it needs to be preprocessed before any further interpretation.

Data Cleaning and Data Preprocessing

In order to prepare the data, non-relevant features or columns needs to be dropped. In this case, “SEVERITYCODE”, “ADDRTYPE”, “WEATHER”, “ROADCOND”, “LIGHTCOND”, “SPEEDING” are selected for future data preprocess. These columns are selected based on the characteristics of the accident. Environmental characteristics such as weather condition, road condition and light condition are likely to affect the output. Address types such as intersection, block and alley determine where is likely to occur accident. Lastly, driving behavior such as speeding also plays a big role in car accidents. However, due to the limited data points in “SPEEDING”, this column will not be taken into account in machine learning model development. After relevant columns are selected, null values from the data set need to be identified and processed. Rows with empty value can be dropped because they only consist of less than 4% of the total data.

At this point, the data is highly imbalanced and non-standardized. The target variable of this model is the severity code, which has 130642 data points for code 1 and 56883 for code 2 after data cleaning. Imbalanced dataset may bias the model if not accounted for. To create a less biased model, resampling method is used to balance the dataset. Detailed process is shown below.



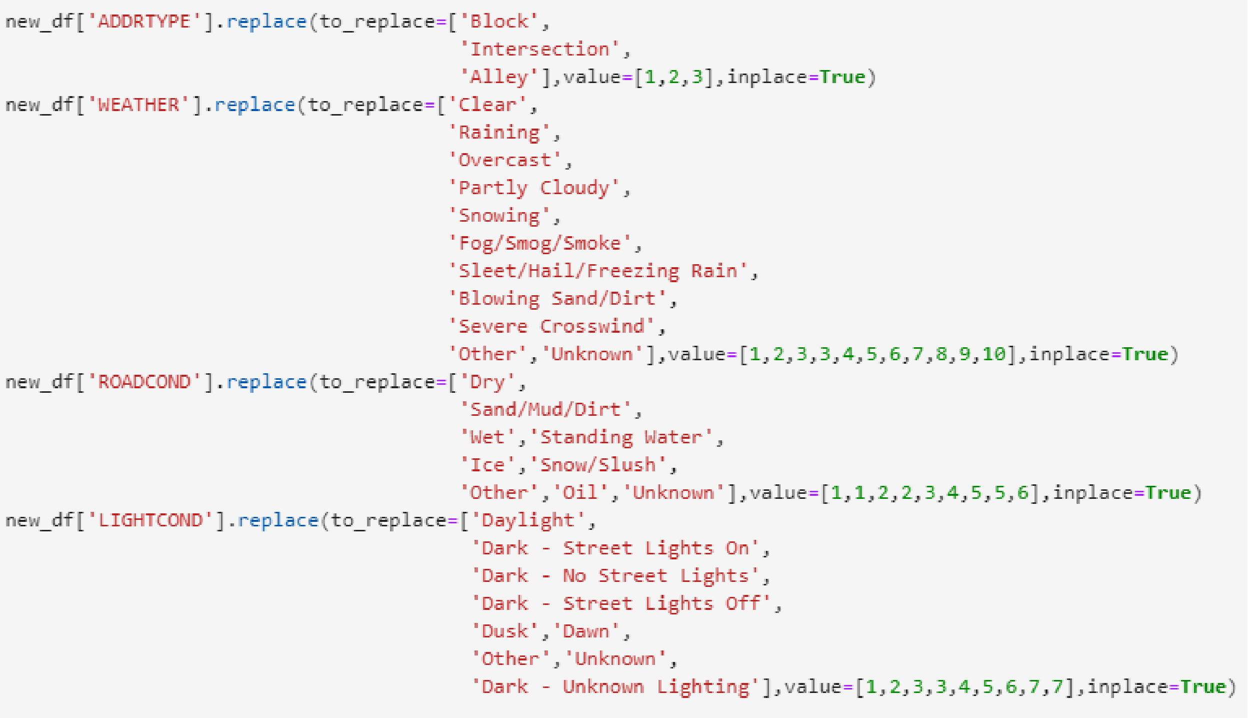
```
from sklearn.utils import resample

df_maj = df[df.SEVERITYCODE==1]
df_min = df[df.SEVERITYCODE == 2]
resample_df = resample(df_maj,replace=False,n_samples=56883,random_state=123)

new_df = pd.concat([resample_df,df_min])
new_df.SEVERITYCODE.value_counts()
```

```
2    56883
1    56883
Name: SEVERITYCODE, dtype: int64
```

Current dataset contains categorical values in each feature. Machine learning models requires numerical data. Therefore, categorical values need to be converted into numerical data. Categorical values in each column of the four features selected were replaced with numbers. For example, each entry in the “ADDRTYPE” column (e.g. “"Block”, “Intersection”, “Alley”) were replaced with “1”, “2” and “3” respectively as shown below.

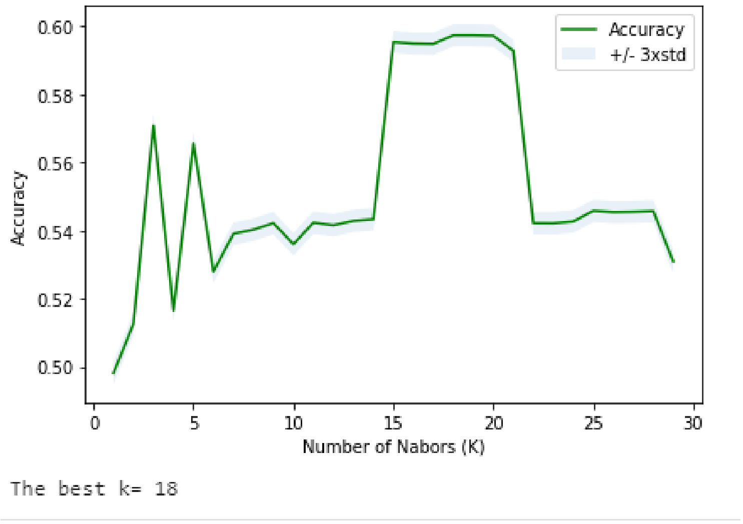


```
new_df['ADDRTYPE'].replace(to_replace=['Block',
                                       'Intersection',
                                       'Alley'],value=[1,2,3],inplace=True)
new_df['WEATHER'].replace(to_replace=['Clear',
                                      'Raining',
                                      'Overcast',
                                      'Partly Cloudy',
                                      'Snowing',
                                      'Fog/Smog/Smoke',
                                      'Sleet/Hail/Freezing Rain',
                                      'Blowing Sand/Dirt',
                                      'Severe Crosswind',
                                      'Other','Unknown'],value=[1,2,3,3,4,5,6,7,8,9,10],inplace=True)
new_df['ROADCOND'].replace(to_replace=['Dry',
                                       'Sand/Mud/Dirt',
                                       'Wet','Standing Water',
                                       'Ice','Snow/Slush',
                                       'Other','Oil','Unknown'],value=[1,1,2,2,3,4,5,5,6],inplace=True)
new_df['LIGHTCOND'].replace(to_replace=['Daylight',
                                        'Dark - Street Lights On',
                                        'Dark - No Street Lights',
                                        'Dark - Street Lights Off',
                                        'Dusk','Dawn',
                                        'Other','Unknown',
                                        'Dark - Unknown Lighting'],value=[1,2,3,3,4,5,6,7,7],inplace=True)
```

Model Development and Evaluation

In order to develop a model for predicting accident severity, the re-sampled, processed dataset was split in to testing and training sets (containing 20% and 80% of the samples, respectively) using the train and test split method. There are four machine learning model have been constructed and evaluated.

K-Nearest Neighbor method is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure. Establishing the value of K is very essential for model development because it dominates the accuracy of the model. KNN model were built for k=1-30 using the kNeighborsClassifier function. As a result, the accuracy of the model is at it best when K equals to 18, at which the model correctly predicts the accident severity code 59% of the time as shown below.



```
The best k= 18
```

Decision tree models breaks down the data set into smaller subsets. An incremental decision tree model is built according to the “entropy” criterion and allowed to run until convergence. SVM models seek to separate data based on different values of the target variable by mapping the dataset to a higher-dimension space and identifying the support vectors which best-describe the hyper planes that most effectively partition the data. C-Support Vector Classification method has been used with “rbf” kernel employed. Logistic regression is also used to classify accident outcomes based on the features. In this case, a logistic regression model was trained using an 0.01 inverse regularization strength number with a linear solver. After the development of the four machine learning models, model accuracy will be determined with confusion matrix, Jaccard score, precision, recall and f1- score methods for each model.

Results

F1 Score: f1 score is a measure of a test’s accuracy. It is calculated from the precision and recall of the test, where the precision is the number of correctly identified positive results divided by the number of all positive results, including those not identified correctly, and the recall is the number of correctly identified positive results divided by the number of all samples that should have been identified as positive.

Jaccard Score: The Jaccard similarity index compares members for two sets to see which members are shared and which are distinct. It's a measure of similarity for the two sets of data, with a range from 0% to 100%. The higher the percentage, the more similar the two populations.

Precision: Precision is the number of relevant documents retrieved by a search divided by the total number of documents retrieved by that search.

Recall: Recall is the number of relevant documents retrieved by a search divided by the total number of existing relevant documents

A summarized result table is shown below. For detailed results and confusion matrix, please see appendix.

Model	f1-score	Jaccard Score	Severity Code	Precision	Recall
KNN	0.592	0.597	1	0.58	0.71
			2	0.63	0.49
Decision Tree	0.594	0.600	1	0.58	0.73
			2	0.64	0.47
SVM	0.595	0.601	1	0.58	0.73
			2	0.64	0.47
Logistic Regression	0.599	0.592	1	0.58	0.73
			2	0.64	0.47

Conclusion

From exploratory data analysis, it can be concluded that weather condition has the highest impact on the severity code. Four machine learning models have been trained and evalueated for their accuracy: 1) K-Nearest Neighbours, 2) Decision Tree, 3) Support Vector Machine, 4)Logistic Regression. All four models perform similarly.

However, the accuracy of the models can be further optimized. The models could have performed better if:

(1) different features can be selected for the models,

(2) different criterion and maximum depth of the decision tree can be selected,

(3) different kernel of SVM model can be selected,

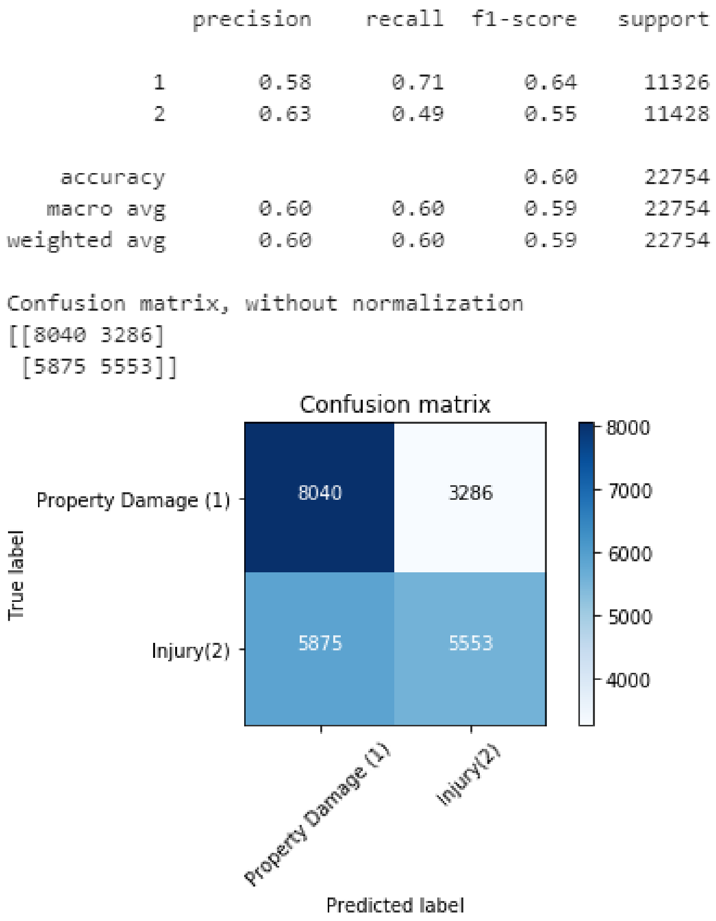
(4) different c value and solver of logistic regression model can be selected,

(5) a balanced dataset is provided for the target variable

(6) complete dataset is provided for feature variables such as Speeding and DUI

Appendix

KNN Confusion Matrix



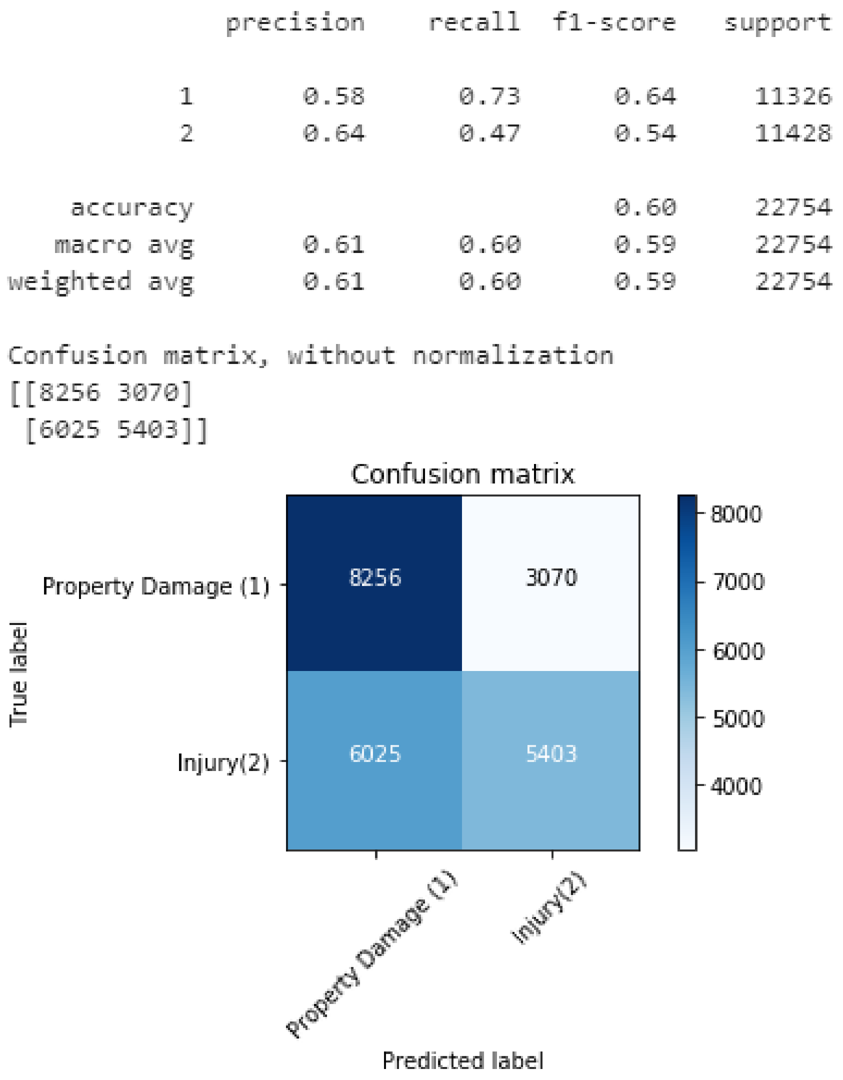
```
              precision    recall  f1-score   support

           1       0.58      0.71      0.64     11326
           2       0.63      0.49      0.55     11428

    accuracy                           0.60     22754
   macro avg       0.60      0.60      0.59     22754
weighted avg       0.60      0.60      0.59     22754

Confusion matrix, without normalization
[[8040 3286]
 [5875 5553]]
```

Decision Tree Matrix



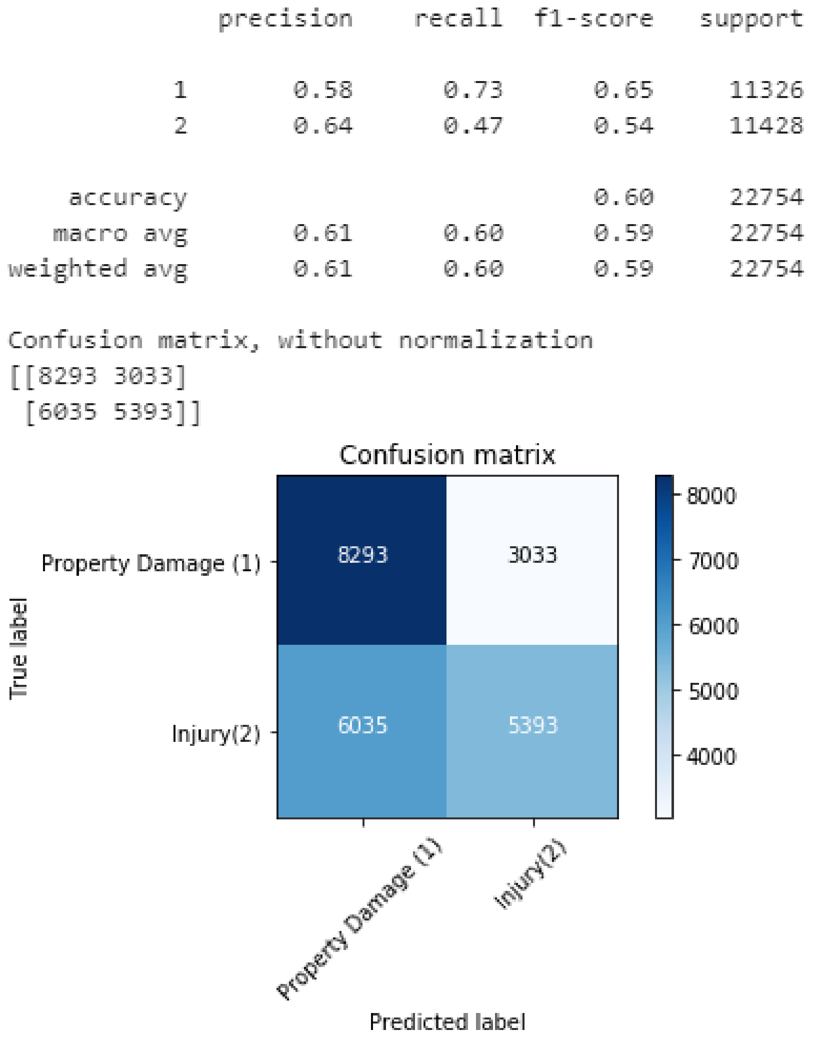
```
              precision    recall  f1-score   support

           1       0.58      0.73      0.64     11326
           2       0.64      0.47      0.54     11428

    accuracy                           0.60     22754
   macro avg       0.61      0.60      0.59     22754
weighted avg       0.61      0.60      0.59     22754

Confusion matrix, without normalization
[[8256 3070]
 [6025 5403]]
```

SVM Matrix



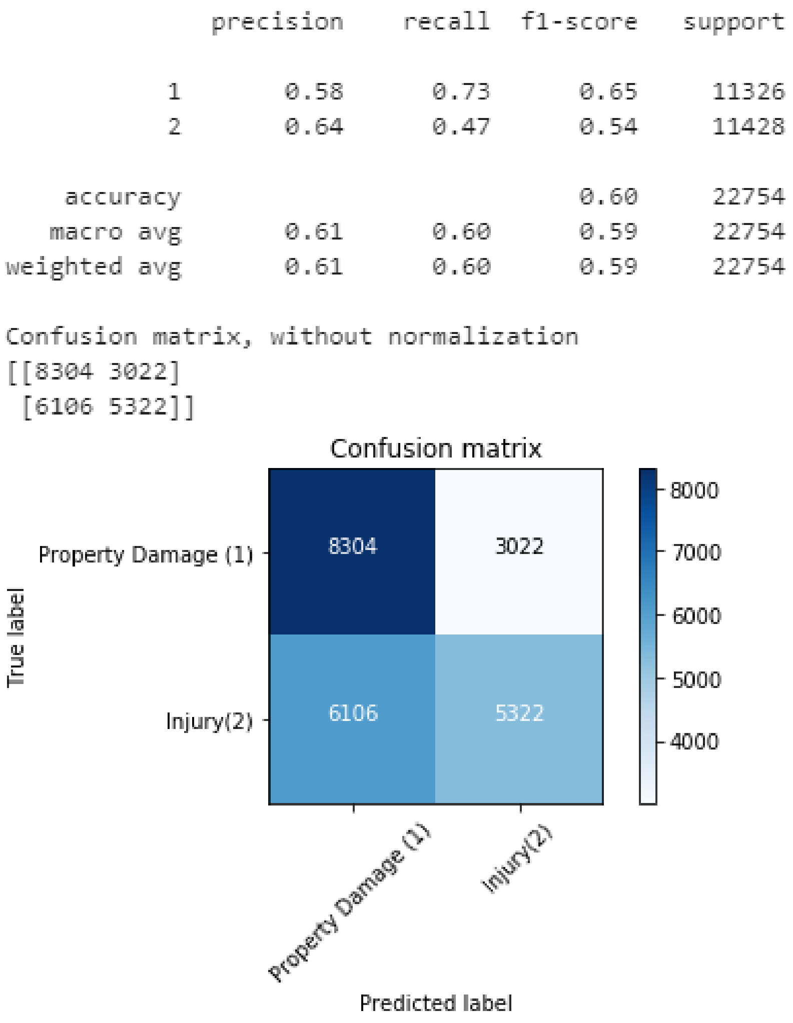
```
              precision    recall  f1-score   support

           1       0.58      0.73      0.65     11326
           2       0.64      0.47      0.54     11428

    accuracy                           0.60     22754
   macro avg       0.61      0.60      0.59     22754
weighted avg       0.61      0.60      0.59     22754

Confusion matrix, without normalization
[[8293 3033]
 [6035 5393]]
```

Logistic Regression Matrix



```
              precision    recall  f1-score   support

           1       0.58      0.73      0.65     11326
           2       0.64      0.47      0.54     11428

    accuracy                           0.60     22754
   macro avg       0.61      0.60      0.59     22754
weighted avg       0.61      0.60      0.59     22754

Confusion matrix, without normalization
[[8304 3022]
 [6106 5322]]
```