Finding the Safest Route

M. Bruggeling

September 28, 2020

Introduction

Background

Seventy-seven percent of smartphone owners use navigation apps like Google Maps or Waze regularly to reach their destination. Navigation apps are so popular because they make navigating more convenient and help save time and resources by actively guiding the user along the quickest route. However, despite their benefits, navigation apps may also have negative impacts.

Problem

Experts have indicated that users can end up not only following the quickest route but potentially also a more dangerous route when blindly relying on their navigation apps. Consequently, accidents and serious injury may occur.

Interest

To provide a solution for this issue and attract new users, an undisclosed navigation app is interested in offering the safest route option. When this option is selected, a prediction is made about where injury causing accident are more likely to occur given the junction type, road, weather, and light conditions. Subsequently, these areas can be circumvented.

Data

Data source

The used data source is the “Example Dataset” which can be downloaded in Week 1.

Feature selection

The selected features are factors that could have influenced an injury causing accident and can be determined before choosing a certain route. Driver-related factors such as speeding or being under the influence of alcohol or drugs are not taken into account, because it is assumed that users who are interested in following a safe route do not exhibit such behavior. The remaining factors (i.e. independent variables) are:

- 'JUNCTIONTYPE': shows the type of junction where a collision happened (e.g. ‘at intersection’).
- 'ROADCOND': indicates the condition of the road at the time of collision (e.g. wet).
- 'LIGHTCOND': represents the light conditions when a collision occurred (e.g. ‘dark - street lights on’).
- 'WEATHER': describes the weather conditions at the time of collision (e.g. overcast).

Furthermore, the dependent variable is:

- 'SEVERITYCODE': indicates whether an accident is injury causing (category 2) or an accident with property damage only (without injury) (category 1).

Pre-processing

Three pre-processing steps were taken:

1. Remove rows with missing values
2. Change dependent variable categories 1 and 2 to values 0 and 1
3. Convert categories in independent variables to separate indicator variables (i.e. dummy variables)

Methodology

Exploratory data analysis

To gain an overview of the data and their main characteristics, the different values (0 and 1) for each variable were counted (Table 1) and a correlation matrix was created (see notebook).

Table 1: Value counts per variable

Variable	0	1
SEVERITYCODE	126527	56669
JUNCTIONTYPE_At Intersection (but not related to intersection)	181139	2057
JUNCTIONTYPE_At Intersection (intersection related)	121955	61241
JUNCTIONTYPE_Driveway Junction	172676	10520
JUNCTIONTYPE_Mid-Block (but intersection related)	160843	22353
JUNCTIONTYPE_Mid-Block (not related to intersection)	96340	86856
JUNCTIONTYPE_Ramp Junction	183034	162
JUNCTIONTYPE_Unknown	183189	7
ROADCOND_Dry	60930	122266
ROADCOND_Ice	182017	1179
ROADCOND_Oil	183136	60
ROADCOND_Other	183073	123
ROADCOND_Sand/Mud/Dirt	183129	67
ROADCOND_Snow/Slush	182216	980
ROADCOND_Standing Water	183087	109
ROADCOND_Unknown	171542	11654
ROADCOND_Wet	136438	46758
LIGHTCOND_Dark - No Street Lights	181733	1463
LIGHTCOND_Dark - Street Lights Off	182038	1158
LIGHTCOND_Dark - Street Lights On	135603	47593
LIGHTCOND_Dark - Unknown Lighting	183185	11
LIGHTCOND_Dawn	180742	2454
LIGHTCOND_Daylight	69224	113972
LIGHTCOND_Dusk	177415	5781
LIGHTCOND_Other	182985	211
LIGHTCOND_Unknown	172643	10553
WEATHER_Blowing Sand/Dirt	183147	49
WEATHER_Clear	74033	109163
WEATHER_Fog/Smog/Smoke	182638	558
WEATHER_Other	182447	749
WEATHER_Overcast	155988	27208
WEATHER_Partly Cloudy	183191	5
WEATHER_Raining	150518	32678
WEATHER_Severe Crosswind	183171	25
WEATHER_Sleet/Hail/Freezing Rain	183084	112
WEATHER_Snowing	182314	882
WEATHER_Unknown	171429	11767

Machine learning algorithms

Because the aim is to classify locations as potentially injury causing or not (property damage only) a classification algorithm is used. Multiple classification algorithms were applied: K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Logistic Regression, (LR) and Decision Tree (DT).

Evaluation

First, the dataset was split in a train (80%) and test set (20%). Next, each algorithm was trained, tested, and evaluated using the Jaccard, LogLoss, and F1-scores.

Application

The best performing model was applied to predict and map injury causing accident locations using the test data.

Results

The best performing model is the K-Nearest Neighbors model using a number of neighbors of 2 (Figure 1). The evaluated accuracies of the model are 66.85% (Jaccard) and 58.77% (F1-score). Furthermore, other models perform quite similar (Table 2). Finally, a map was created that shows the predicted injury causing accident locations (Figure 3).

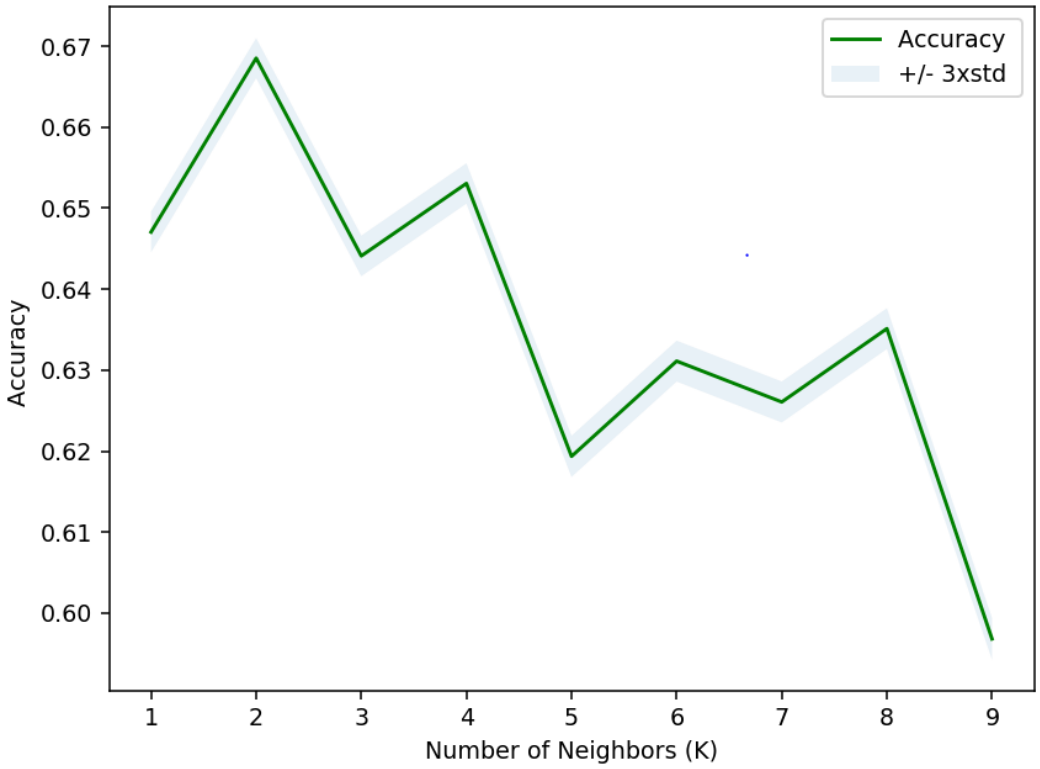

Figure 1: Accuracy of K Nearest-Neighbor model for different numbers of neighbors

Table 2: Model performance evaluation

Algorithm	Jaccard	F1-score	LogLoss
KNN	0.66853	0.58771	NA
DT	0.68835	0.56128	NA
SVM	0.68835	0.56133	NA
LR	0.68835	0.56128	0.58822

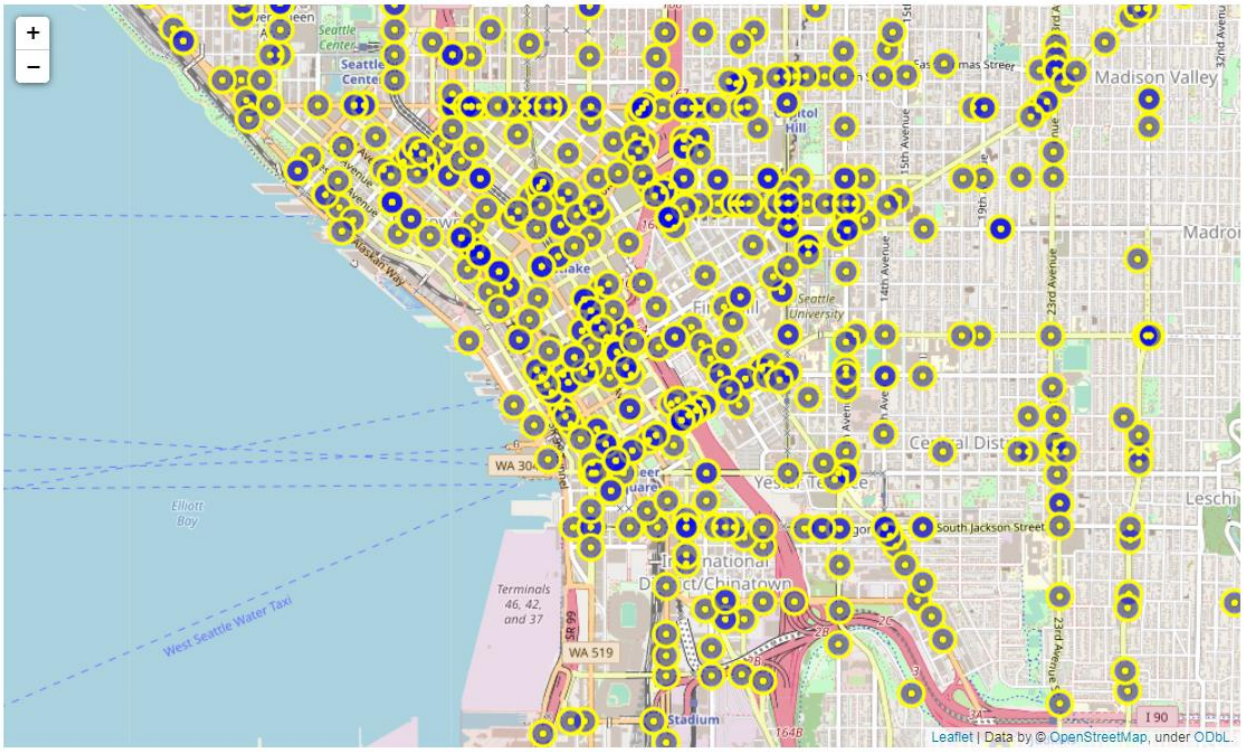

Figure 2: Predicted injury causing accident locations in a part of Seattle

Conclusion

Using a KNN model, a navigation app can determine where injury causing accidents may occur with a success rate of 58.77-66.85%. These locations can be predicted in advance of a user's drive by collecting and preparing infrastructural information (road conditions and and junction types) and combining it with the expected light and weather conditions (forecast) along potential routes. It follows that, by offering a route that circumvents these location, the safest route option can be found.