



Data^x

Traffic Risk Group

Building a real-time map to identify the safety of surrounding roads

Bilai, Nga Pui, Zhao, Saif and Yiran

Top Technical Requirements

- Retrieve and update the movement of risky drivers using real time data
- Machine learning algorithm - no one has had previous experience in building algorithms
- Build an app that has some mapping and speech recognition functions
- Pull real time data
- Color code routes to depict the intensity of risk

Top User Requirements

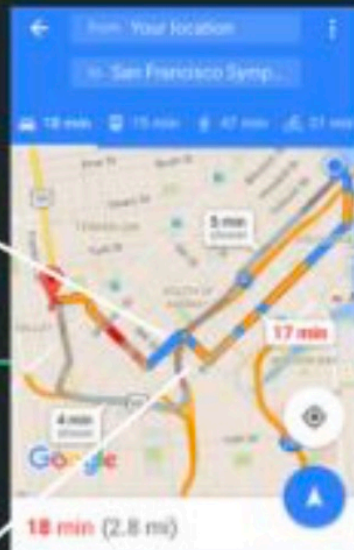
- Traffic map that uses real time data to offer alternative routes to reduce risk
- Speech recognition that processes the driver's answers/commands, i.e. whether or not to switch to another route when prompted /warned about a dangerous zone ahead
- Interactive and easy to read and navigate interface for drivers to conveniently switch routes when driving
- Color-coding system that speaks to the user about the intensity of risk for each route by translating percentages to colors, i.e. what does 26% less risk exactly mean?

Final User Interface

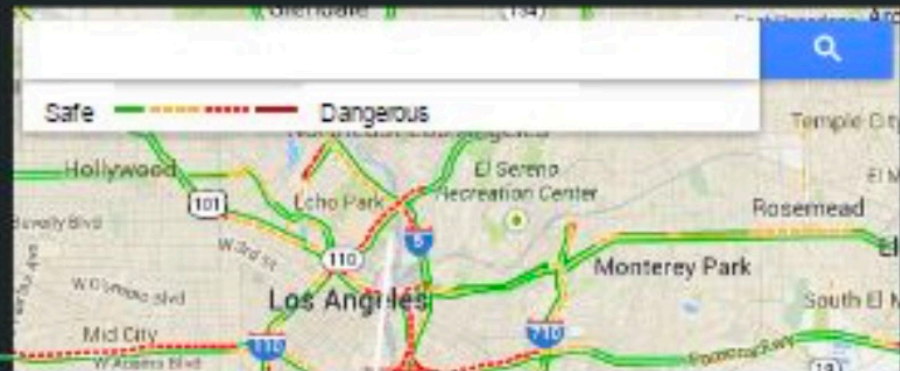
Traffic Risk Model

Bilal El-Halabi, Nga Pui Leung, Zhao Sheng, Yiran Chen, Syed Saif Nizam

Propose routes that are not not much longer but decrease risk level significantly



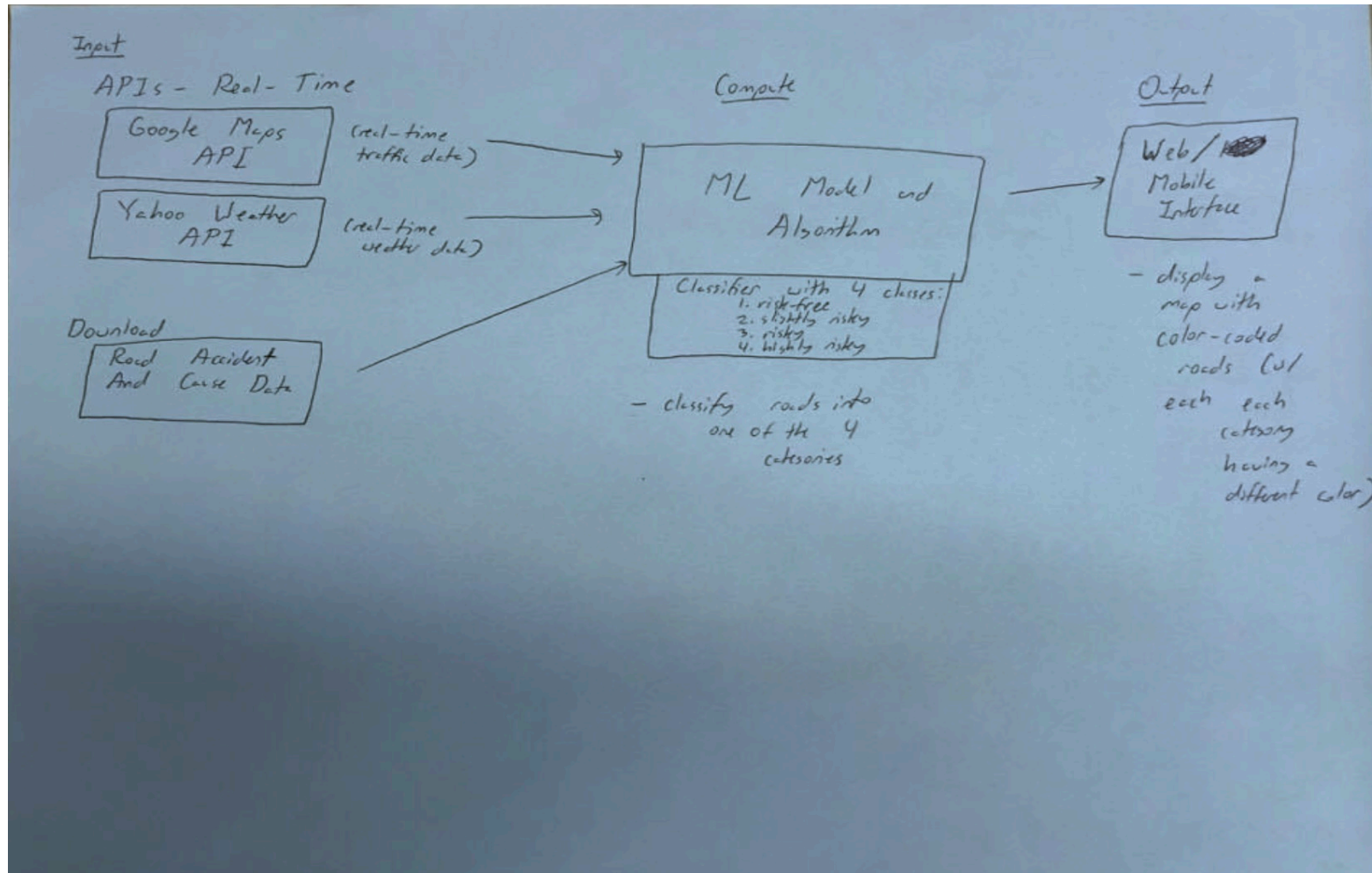
Interactive and easy to read and navigate interface for drivers to conveniently switch routes when driving



Highlights and color codes sections of roads to show level of riskiness of strips of road at current time

Uses voice recognition to process the driver's answer when prompted: "Do you want to switch to a route that is only 4 minutes slower but reduces your risk of an accident by 26%."

Sample Structure



Week 7 Traffic Risk

- Find data sets for weather, traffic, and potentially driver records (Nga Pui)
- Find good tutorials for building apps, ideally also with real-time data involved (Saif, Zhao)
- Talk to professors experienced with traffic infrastructure and begin to verify our user requirements (Bilal)
- Create a survey that we can send out/email to a wide target audience about our app idea and useful features (Yiran)
- ...

Code demonstration

In [1]:

```
import pywapi
```

In [2]:

```
ny=pywapi.get_weather_from_weather_com('10001')
```

In [3]:

```
ny['current_conditions']
```

Out[3]:

```
{'barometer': {'direction': 'rising rapidly', 'reading': '1006.43'},
 'dewpoint': '-10',
 'feels_like': '3',
 'humidity': '30',
 'icon': '34',
 'last_updated': '3/02/17 3:51 PM EST',
 'moon_phase': {'icon': '4', 'text': 'Waxing Crescent'},
 'station': 'Nyc/Central Park, NY, US',
 'temperature': '6',
 'text': 'Fair',
 'uv': {'index': '1', 'text': 'Low'},
 'visibility': '16.1',
 'wind': {'direction': '300', 'gust': '46', 'speed': '14', 'text': 'WNW'}}
```

In [4]:

```
print ("Weather.com says: It is " + ny['current_conditions']['text'].lower() + "
and " + ny['current_conditions']['temperature'] + "°C now in New York.")
```

Weather.com says: It is fair and 6°C now in New York.

In [5]:

```
if int(ny['current_conditions']['wind']['speed'])>15 or float(ny['current_conditions']['visibility'])<10:
    print("Slight driving risk")
elif int(ny['current_conditions']['wind']['speed'])>15 and float(ny['current_conditions']['visibility'])<10:
    print("Moderate driving risk")
else:
    print("No driving risk")
```

No driving risk