

# Accident Prophet: Predicting Road Incidents

## Group 11

Mr. Nitesh Ghimire

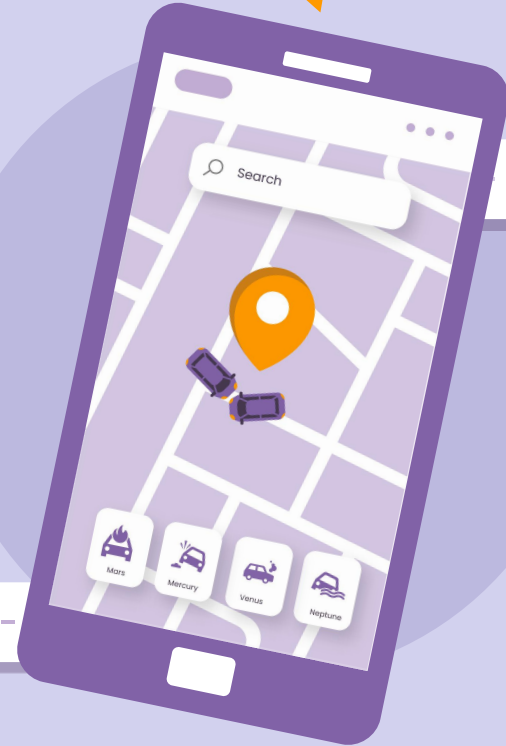
Mr. Gholamreza Izadi

Mr. Tairo Kageyama

Mr. Panithi Sirisatjapipat

Mr. Bakyt Tursaliev

REPORT!





# Table of contents

**01**

**Introduction**

**02**

**Data Wrangling**

**03**

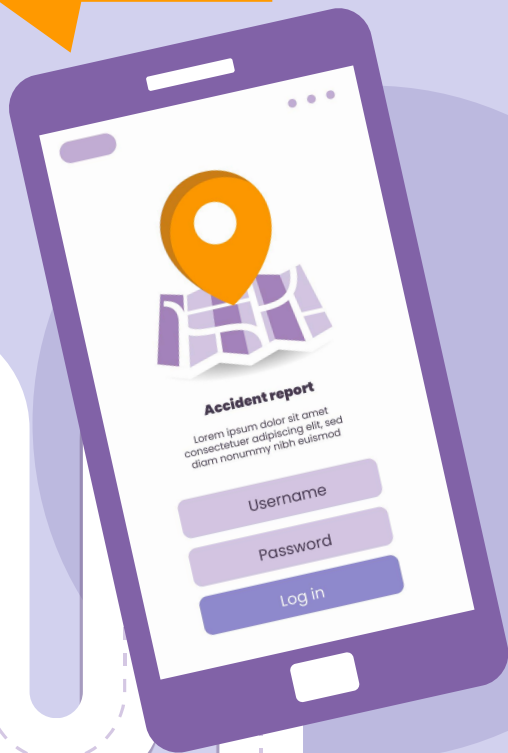
**Modeling**

**04**

**Deployment**

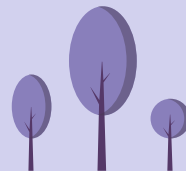


**ACCIDENT!**



**01**

# Introduction

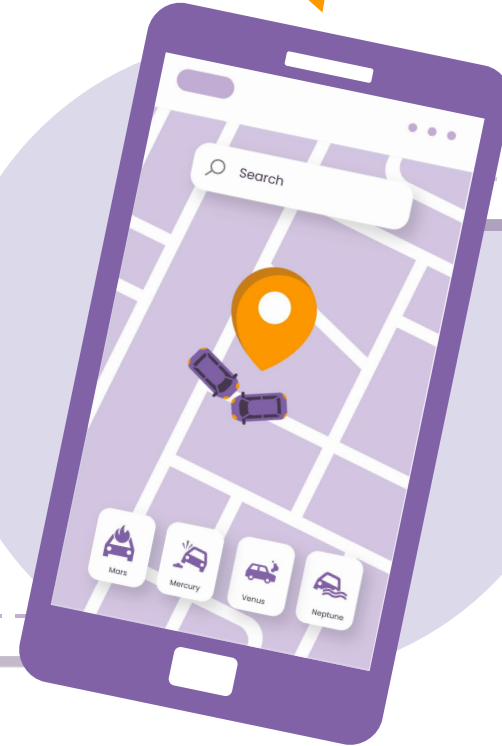




# Introduction

Our product is driven by the belief that data-driven decision-making, coupled with the power of machine learning, can significantly enhance the way we understand and manage traffic accidents. "AccidentProphet" harnesses the potential of machine learning and data analytics to unlock valuable patterns and trends within accident records.

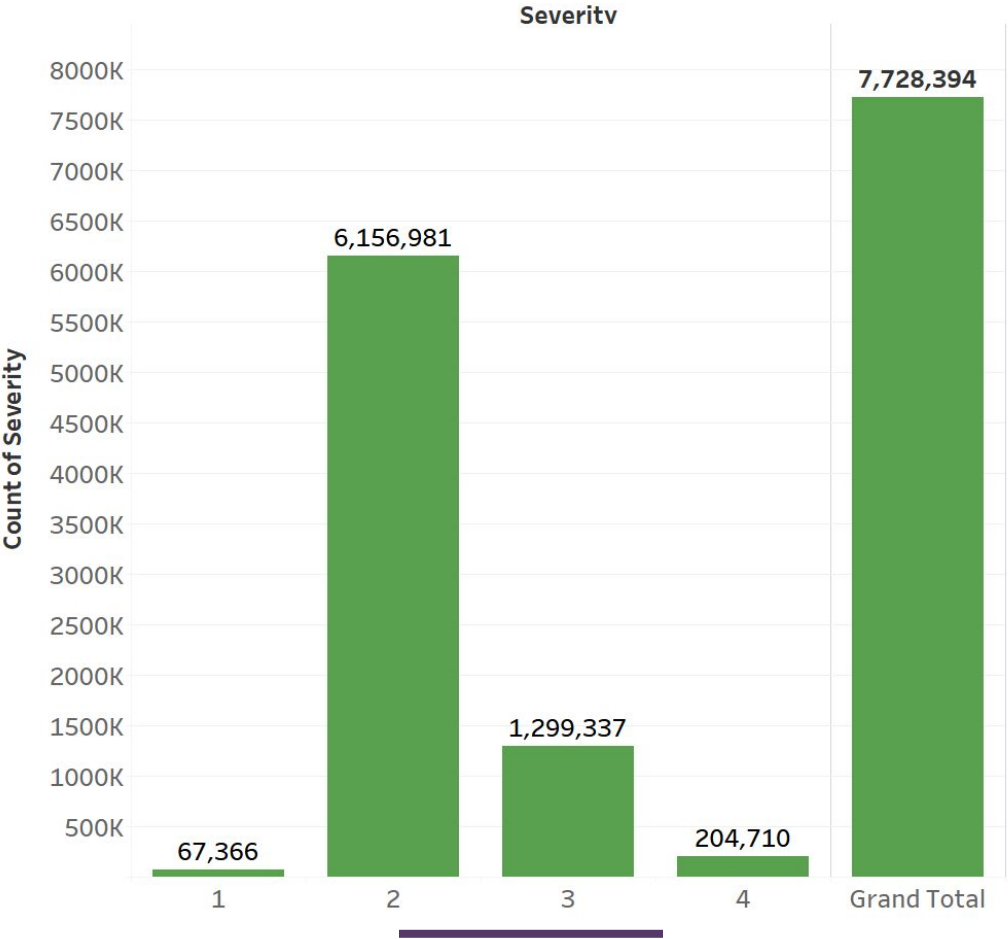
REPORT!



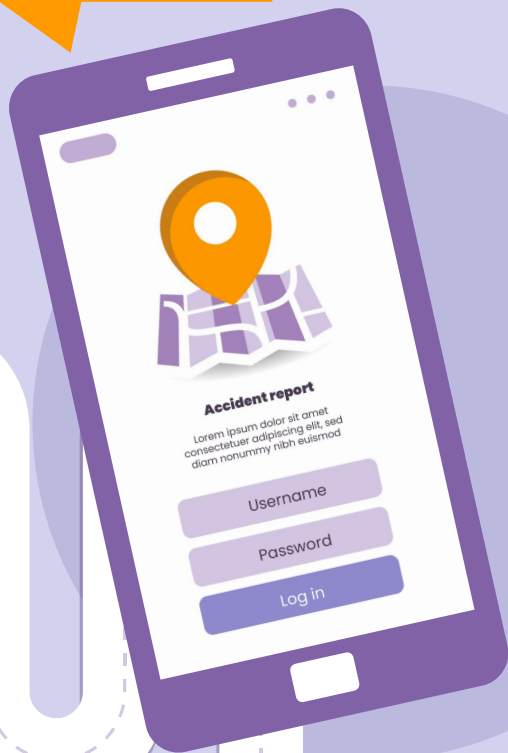
DATA

Accident information		Location		Road specification		Weather		Time	
ID	object	Start_Lat	float64	Amenity	bool	Airport_Code	object	Start_Time	object
Source	object	Start_Lng	float64	Bump	bool	Weather_Timestamp	object	End_Time	object
Severity	int64	End_Lat	float64	Crossing	bool	Temperature(F)	float64	Timezone	object
Distance(mi)	float64	End_Lng	float64	Give_Way	bool	Wind_Chill(F)	float64		
Description	object	Street	object	Junction	bool	Humidity(%)	float64		
		City	object	No_Exit	bool	Pressure(in)	float64		
		County	object	Railway	bool	Visibility(mi)	float64		
		State	object	Roundabout	bool	Wind_Direction	object		
		Zipcode	object	Station	bool	Wind_Speed(mph)	float64		
		Country	object	Stop	bool	Precipitation(in)	float64		
				Traffic_Calming	bool	Weather_Condition	object		
				Traffic_Signal	bool	Sunrise_Sunset	object		
				Turning_Loop	bool	Civil_Twilight	object		
						Nautical_Twilight	object		
						Astronomical_Twilight	object		

DATA

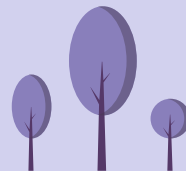


**ACCIDENT!**



**02**

# Data Wrangling



# Data Wrangling



Creating time  
component



Selecting  
records of year  
after 2020



Dropping End\_Lat  
and End\_Long



Removing  
samples with  
missing value



Creating  
Features  
"Delay(min)"



Converting  
Measurement to  
Metric Standard



Recode some variables such  
as weather condition and  
doing one hot encode for  
categorical columns



Recoding  
severity to two  
class attribute

---



# Data Wrangling



Creating count plots, bar plots, and scatter plot for different type of attributes



Describe data columns



Deleting 48 variables



Removing outliers



Creating dataframes X and Y

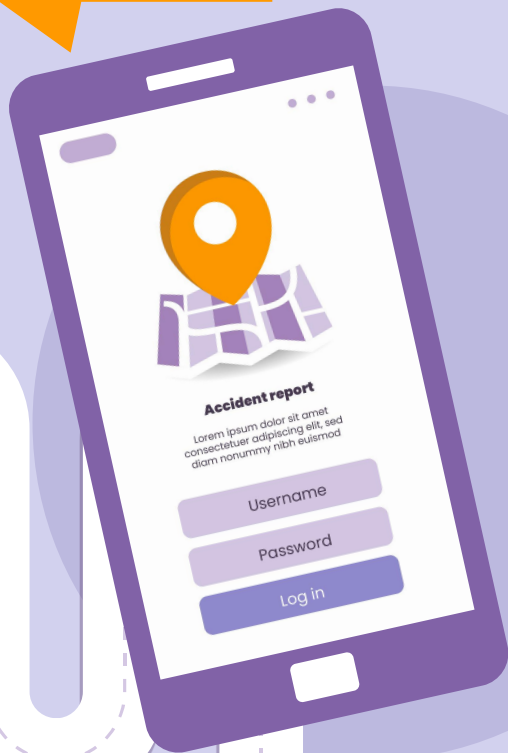


Train-test split



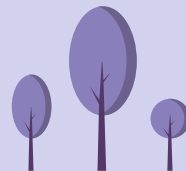
Imbalance fixing

**ACCIDENT!**



**03**

# Modeling



# Modeling

## Features and Label

	Feature name	min	mean	max
1	Start_Lat	24	35	49
2	Start_Lng	-124	-93	-67
3	Humidity(%)	1	64	100
4	Crossing	0	0	1
5	Junction	0	0	1
6	Stop	0	0	1
7	Traffic_Signal	0	0	1
8	Sunrise_Sunset	0	0	1
9	Civil_Twilight	0	0	1
10	Start_Hour	0	12	23
11	IsWeekend	0	0	1
12	Temperature(C)	-29	16	44
13	Pressure(cm)	0	74	78
14	Precipitation(cm)	0	0	81
15	Visibility(km)	0	14	16
16	Wind_Speed(kmph)	0	12	88
17	Weather_Bin_Clear	0	0	1
18	Weather_Bin_Cloudy	0	0	1
19	Weather_Bin_Rainy	0	0	1
20	Weather_Bin_Snowy	0	0	1
21	Start_Month_December	0	0	1
22	Start_Month_January	0	0	1



Severity (0,1)

# Modeling

After data preparation, four classification algorithms (Adaboost, xgboost, Random Forest, Logistic Regression) were used. Cross-validation assessed their performance, helping select the best model based on accuracy, precision, recall, and F1-score..



**AdaBoost**

---

**XGBoost**

---

**Random  
Forest**

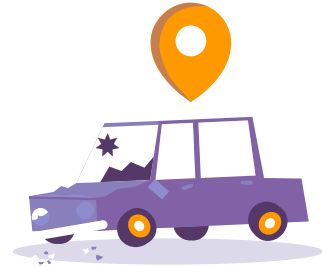
---

**Logistic  
Regression**

---

# Modeling

After data preparation, four classification algorithms (Adaboost, xgboost, Random Forest, Logistic Regression) were used. Cross-validation assessed their performance, helping select the best model based on accuracy, precision, recall, and F1-score..



**AdaBoost**

**XGBoost**

**Random  
Forest**

**Logistic  
Regression**

# Modeling

## Cross Validation



Scores



mean

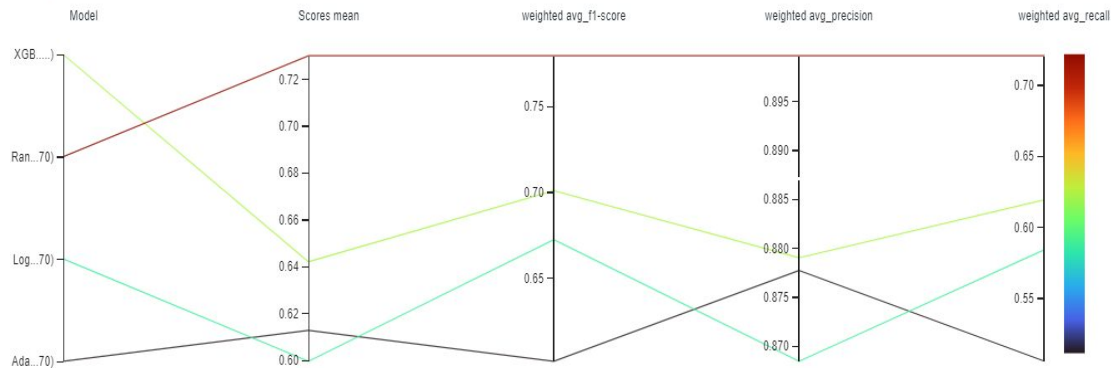
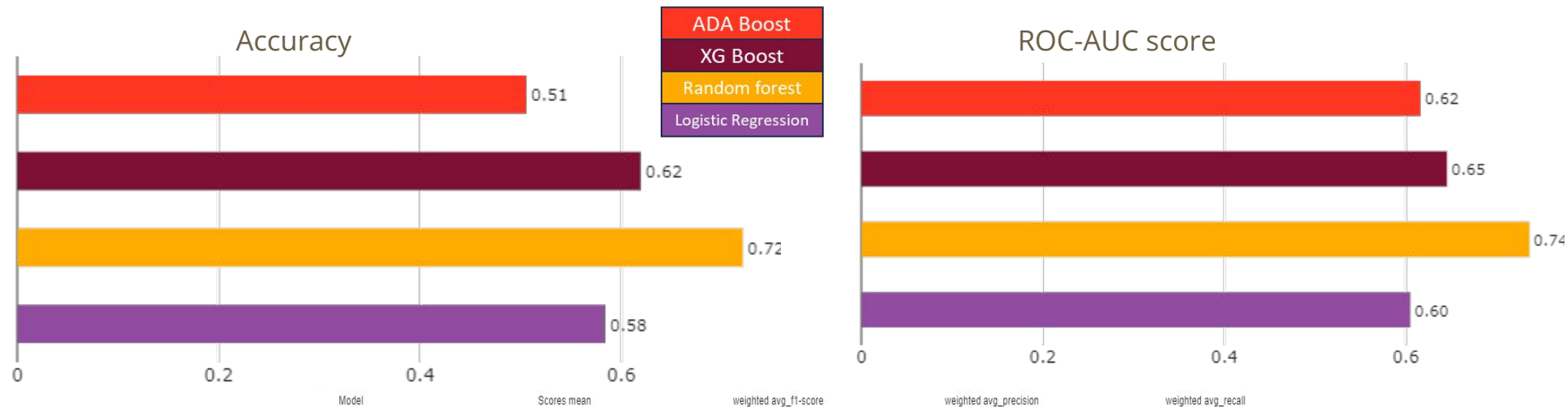


std

AdaBoost	[0.61 0.61 0.61 0.61 0.61]	0.6127	0.001799
XGBoost	[0.64 0.64 0.64 0.64 0.65]	0.6419	0.001856
Random Forest	[0.73 0.73 0.73 0.73 0.73]	0.7299	0.001901
Logistic Regression	[0.6 0.6 0.6 0.6 0.6 ]	0.5995	0.001651

# Modeling

## Cross Validation



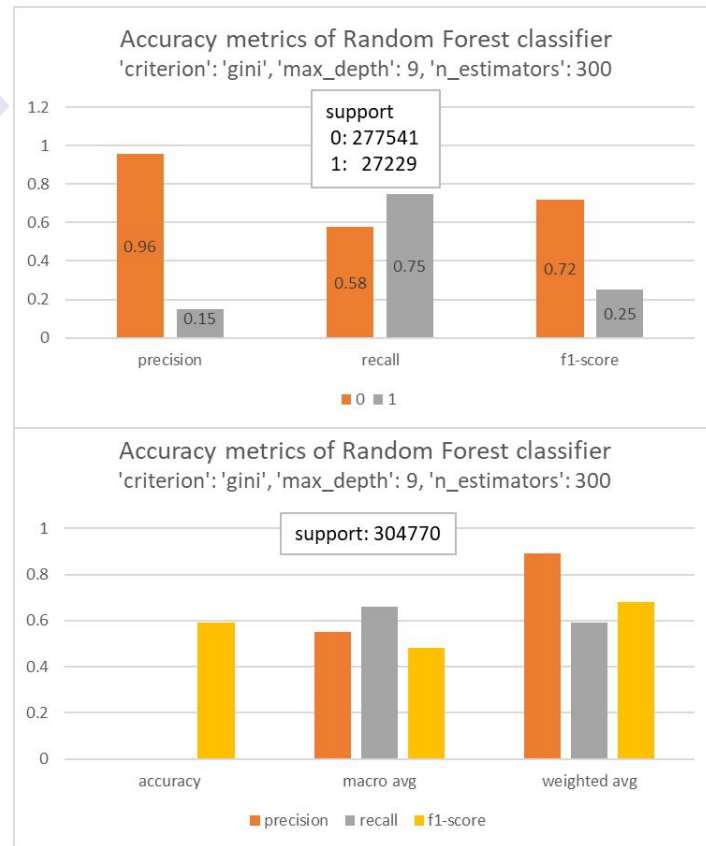
# Modeling

## Grid Search

```
{'criterion': 'gini', 'max_depth': 9, 'n_estimators': 300}
```

	precision	recall	f1-score	support
0	0.96	0.58	0.72	277541
1	0.15	0.75	0.25	27229
accuracy			0.59	304770
macro avg	0.55	0.66	0.48	304770
weighted avg	0.89	0.59	0.68	304770

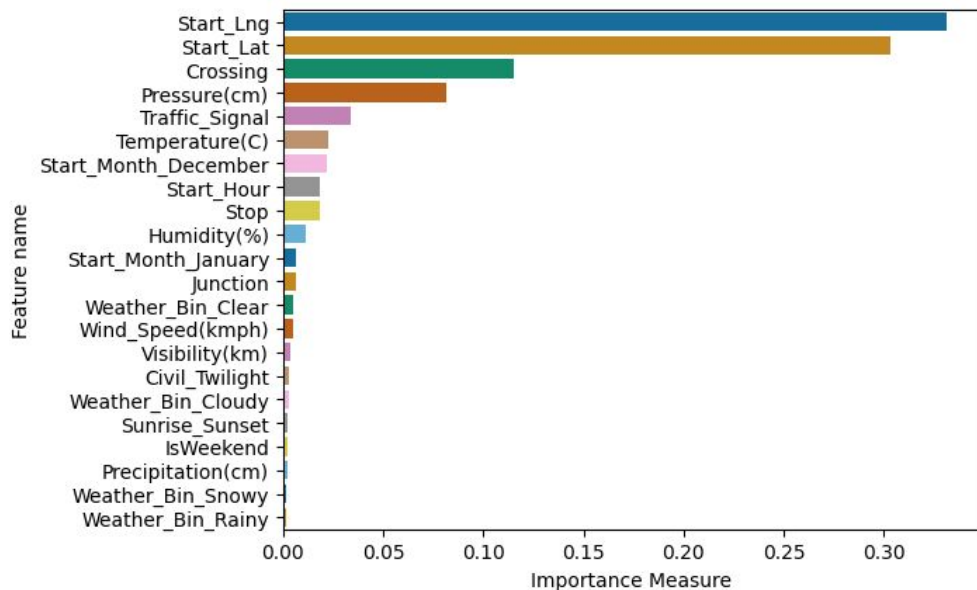
## Modeling Evaluation





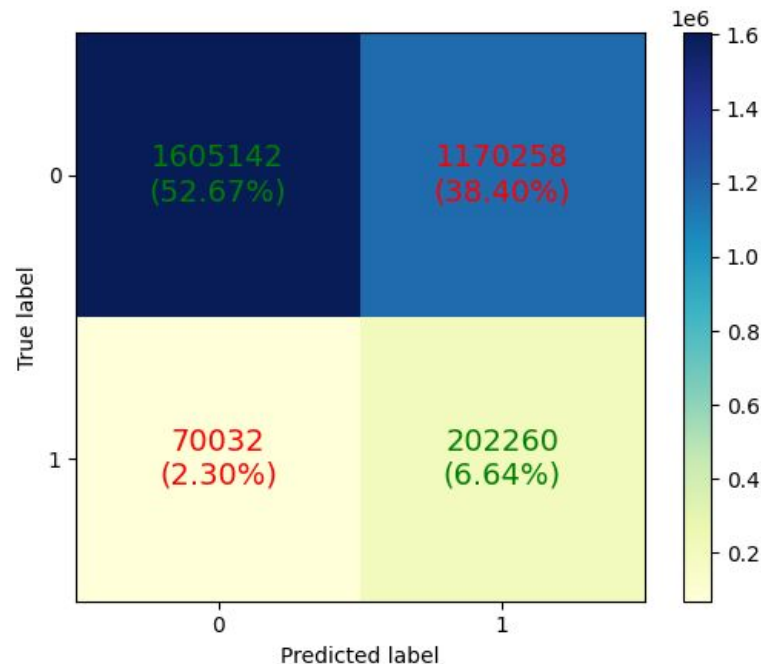
# Modeling

## Feature Importance

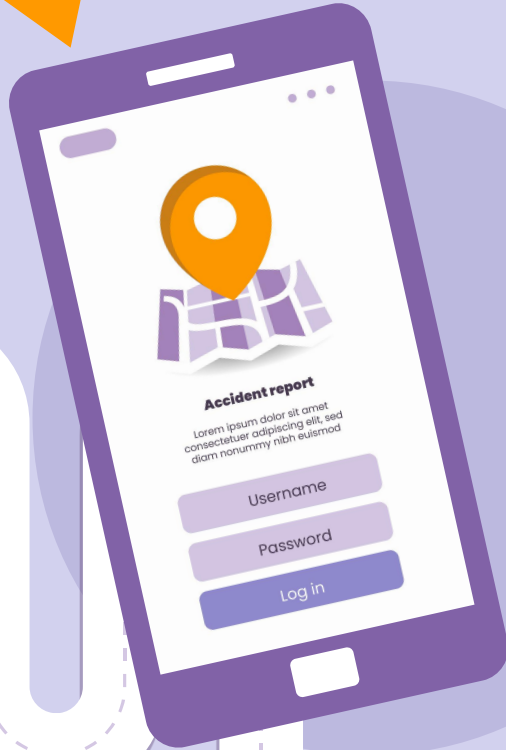


# Modeling

## Inference

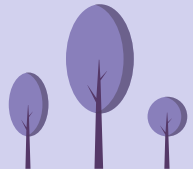


**ACCIDENT!**

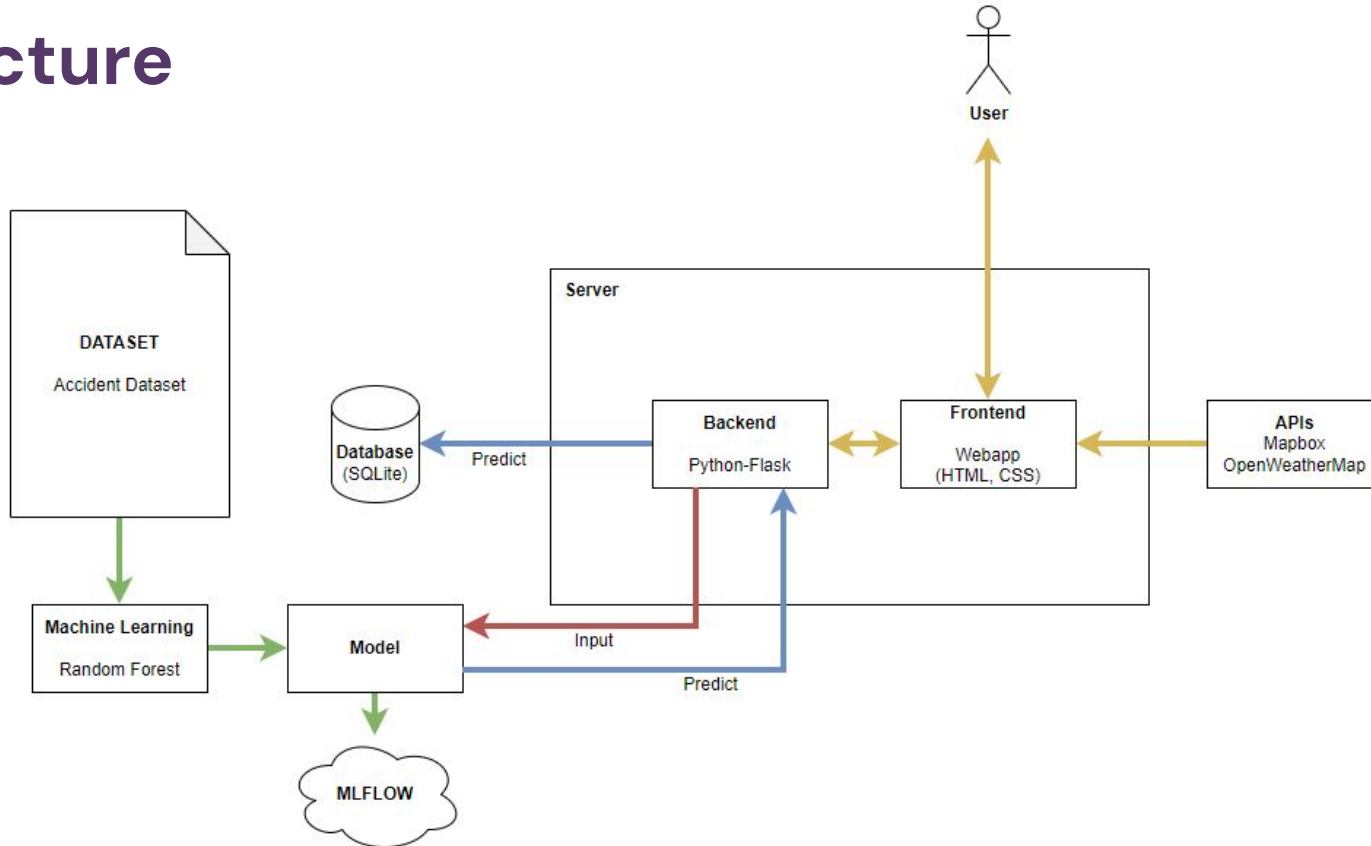


**04**

# Deployment



# Structure

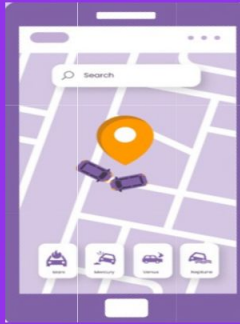


# Home

[Home](#)[About](#)

## Accident Prophet

### Predicting Road Incidents



Our product is driven by the belief that data-driven decision-making, coupled with the power of machine learning, can significantly enhance the way we understand and manage traffic accidents.

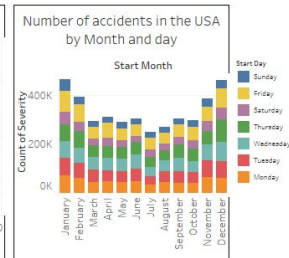
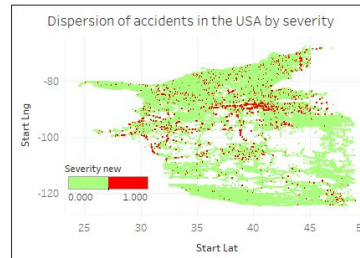
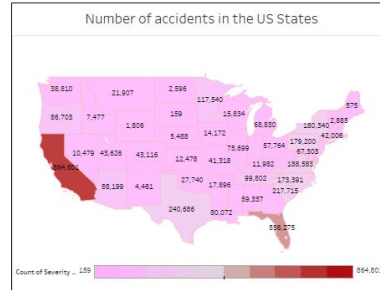
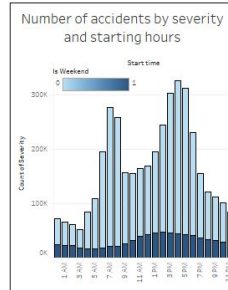
"AccidentProphet" harnesses the potential of machine learning and data analytics to unlock valuable patterns and trends within accident records.

[View Statistics](#)[Get Prediction](#)

# View Statistics

Welcome to AccidentProphet

Predicting Road Incidents

[Home](#)[About](#)

# Predict

Welcome to AccidentProphet

Predicting Road Incidents

Home

About


Execute Python Code

Latitude: Longitude: Visibility(km):

crossing?: Traffic\_Signal?: Any Stops?: Traffic Jams?: Weather: Clear Execute

Get Current Location

Q Enter a location



# Predict

Execute Python Code

Latitude: 40.06865251882823

Longitude: -75.31094236435807

Visibility(km): 0.1

crossing?: Yes

Traffic\_Signal?: No

Any Stops?: Yes

Traffic Jams?: No

Weather: Clear

Execute

[Get Current Location](#)

Welcome to AccidentProphet

Predicting Road Incidents

[Home](#)[About](#)

Execute Python Code

Latitude:

Longitude:

Visibility(km):

crossing?: No

Traffic\_Signal?: No

Any Stops?: No

Traffic Jams?: No

Weather: Clear

Execute

[Get Current Location](#)

Longitude: -75.31094236435807, Latitude: 40.06865251882823

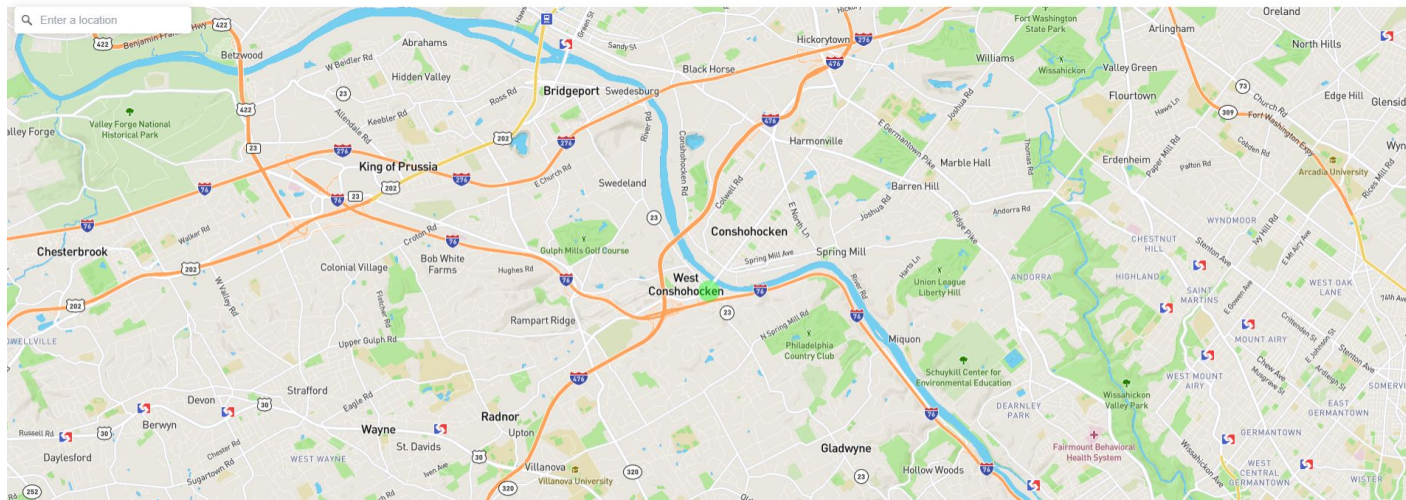
Cross: Yes, Pressure: 101325, Signal: No

Temperature: 15, Stop: Yes, Humidity: 75

Junction: No, Wind Speed: 14.4, Visibility: 0.1

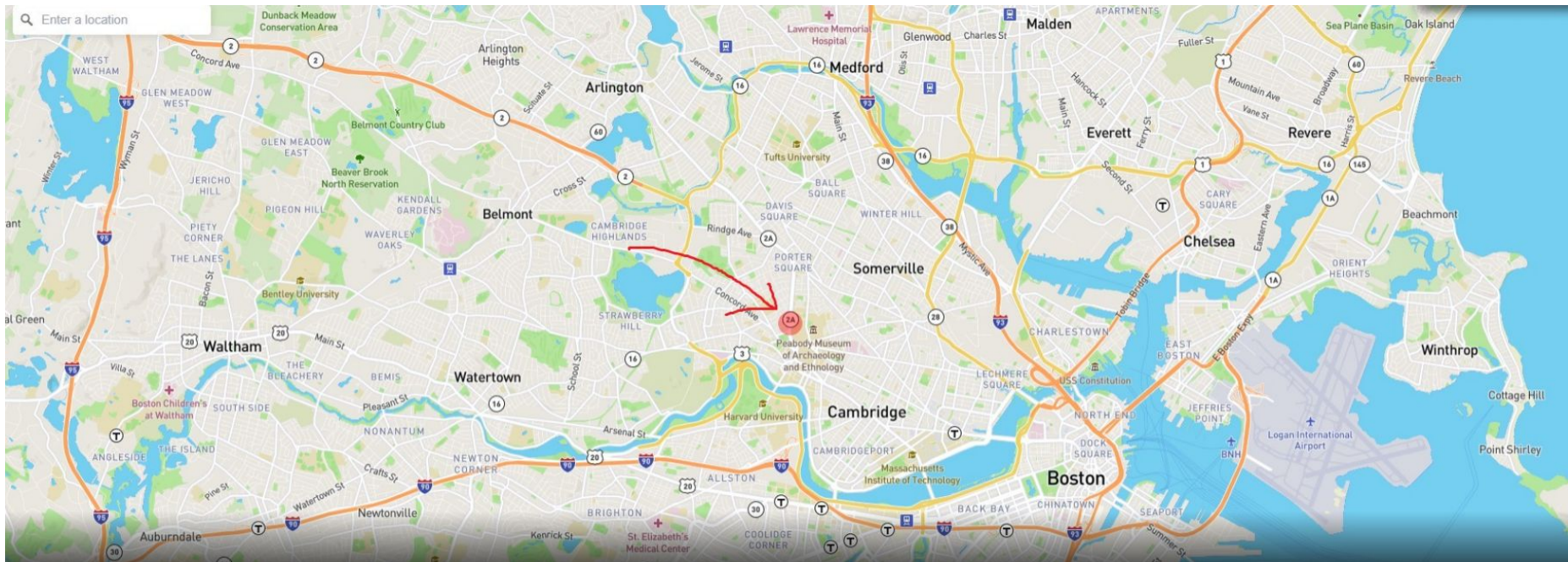
Twilight: No, Sunrise/Sunset: No, Precipitation: 100

Weather: Clear





# Predict







**Thank You So Much!**

