

Crime Analysis Engine

Omkar Gudekar (omkar.gudekar@sjtu.edu)

Sagar Bendale (sagar.bendale@sjtu.edu)

Larkins Carvalho (larkins.carvalho@sjtu.edu)

Nielet Dmello (nielet.dmello@sjtu.edu)

San Jose State University

One Washington Square, San Jose, California -95192-0080, USA

Abstract

Everyday when we travel to places, how often do we know the area around us? Safety is a major concern for everyone. Hence, we need an engine that will aid us to understand how safe a particular area is at any given time. Our Crime Analysis engine provides analysis, visualization and information of the criminal data related to the crimes happening in the area. It will allow the user to see crime hotspots, isolate crimes by type, and identify the days when a particular crime occurs as well as the time.

Keywords- Crime analysis, crime rate.

1. Introduction

Education is the best defense against crime. Hence, we provide the crime analysis engine to provide an analysis about the neighborhood of our interest.

Crime Analysis Engine is a mobile application aimed towards providing a tool to check the safety for any area. The primary goal of the application is to provide relevant and accurate statistical analysis of the crime in any particular area. We have done graphical representations and provided a crime map to highlight the crimes occurring in the area of interest. We have analyzed the data and provided a graphical representation about the crime types, frequency of occurrence, etc. Public can analyze a particular neighborhood to know the crime rate can use the application. This paper aims at giving you a detailed description of our work along with various functionalities implemented in this application.

2. Analysis

We have been through various websites and applications doing crime analysis and generating crime map. However, they do not cater to all requirements. Also, many of those miss a graphical representation. When seen from an end user point, these sites and applications are not so

easy to use. Also, the data used by them is irrelevant and unstructured. Hence, we decided to analyze the crime data of various top cities in the United States and develop a mobile application which will have graphical representation of the data along with a crime map to depict the crime in the area entered by the user. We have provided statistical analysis of the data in following categories:

A. Crime Map

We gathered data from various government databases and other websites, which provide datasets about crimes in various cities of the United States. We transformed the unstructured data to a structured format. We then represent it graphically on Google Maps to plot the crime of the area.

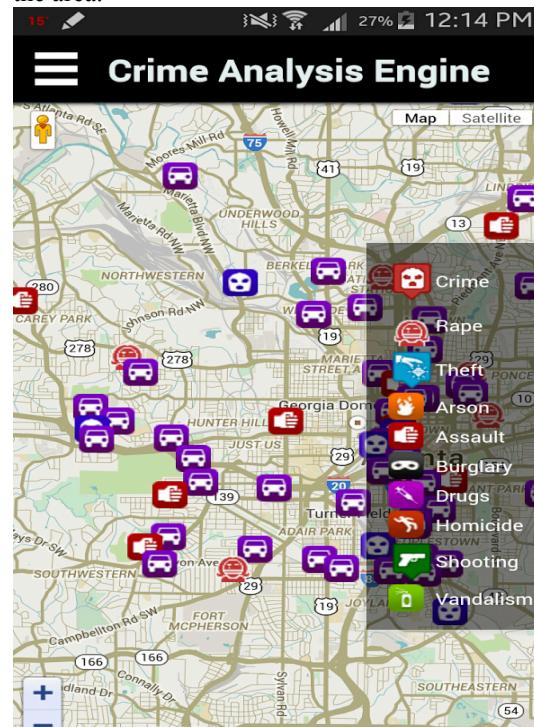


Fig 2.1 Crime plot using Google Maps

B. Crime By Day

Using the data used in A, we have represented the crime by each day of the week for a particular area. Consider a city, which has more crimes happening on weekends than weekdays. Hence, filter it and display it by day.

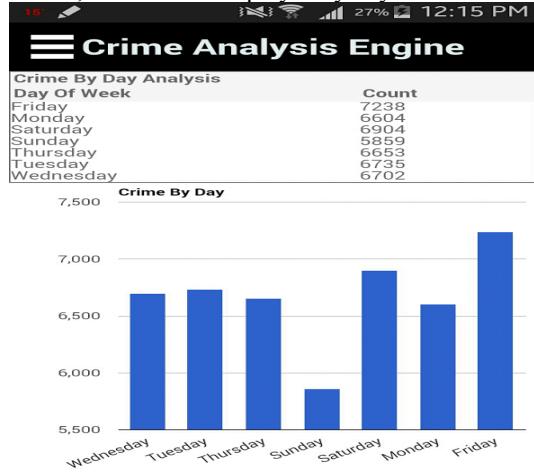


Fig 2.2 Day wise analysis of Crime

C. Crime By Time

In order to make the application even detailed for the user, we further filter the data and provide the crime occurrence by type in a graphical format. Thus, the user can view which crime has occurred the most by a particular time in this section.

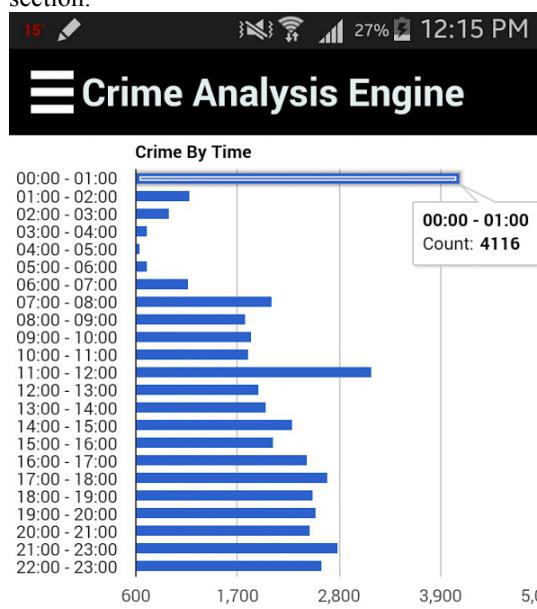


Fig 2.3 Time wise analysis of Crime

D. Crime By Type

We display the crimes by type for a particular area, which the user has inserted. Thus, the user can view the occurrence of crime by its type.

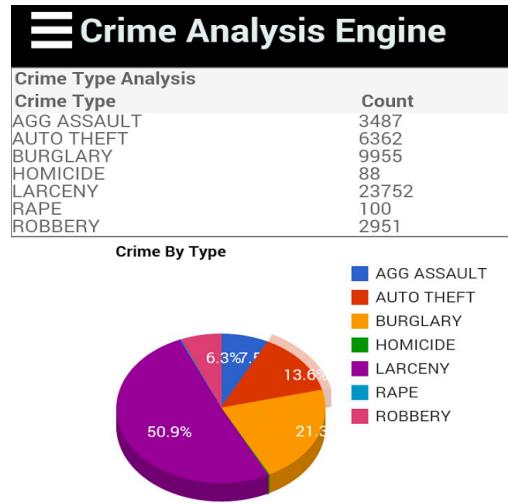


Fig 2.4 Type wise analysis of Crime

E. Solution

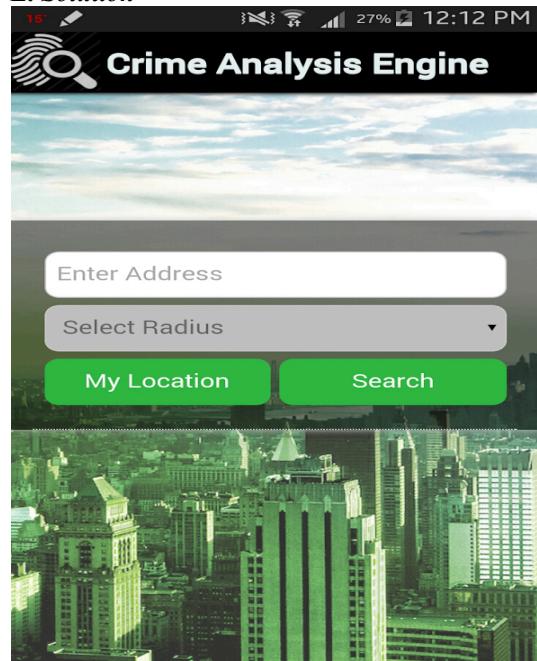


Fig 2.5 Crime Analysis Engine

The user can enter the desired address and select the radius around the area in order to view the crime analysis. Another option provided by us is to allow the user to use his/ her current location.

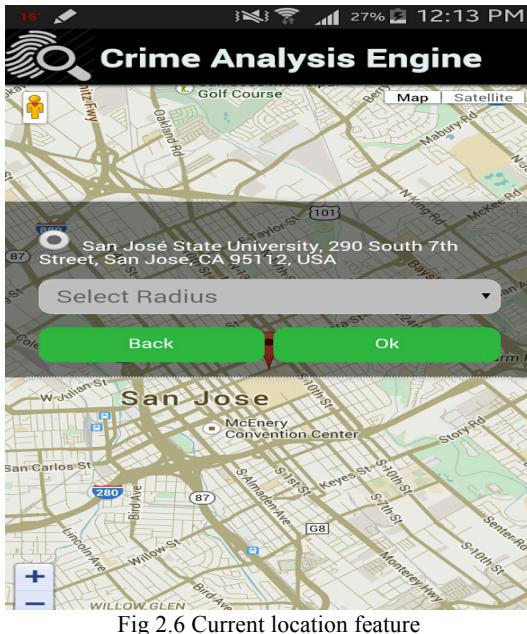


Fig 2.6 Current location feature

Once the user selects the area, he/she can choose from the desired analysis type to view the crime data in graphical format.



Fig 2.7 Various options available in application

3. Architecture

The figure below shows the architecture of our application- ‘Crime Analysis Engine’.

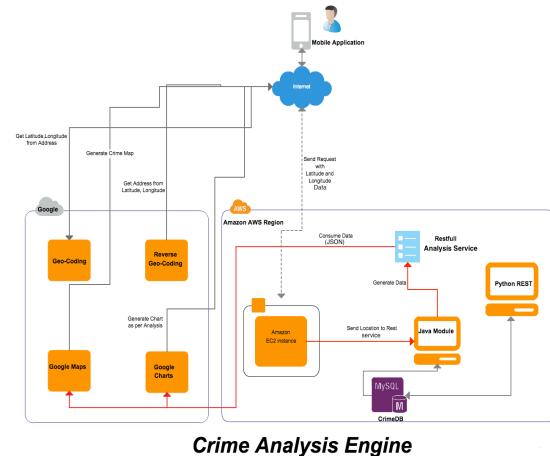


Fig 3.1 Architecture of Crime Analysis Engine

We use the Geo Coding to get the latitude and longitude of a given address. If the user selects the ‘Use my current Location’ option, we use the Reverse Geo Coding. The address is then plotted on the Google Maps and corresponding REST service is called for the type of analysis the user wants to see. Using Google Charts, we provide a graphical representation of our data analysis. We have used MySQL database. The Python REST API is used cleanse the data from the various public data sources.

4. Future Enhancements

We plan to make the application more intuitive by using ways to present data in graphical format by including more type of graphs/charts. Currently, we are working on datasets of selected cities. However, we plan to expand these based on availability and perform an analysis of crime for various cities across the states.

We are trying to take this application on large scale to increase overall awareness about safety and to provide a more accurate, timely crime mapping and analysis.

5. Conclusion

We have successfully analyzed the crime data from various sources to provide an analysis of crime and ensure safety. The design of UI and the use of technology is done keeping in mind the delivery of optimal performance for the end user.

We have done operations on various categories to provide an overall analysis in an accurate manner.

6. Contributions

Omkar Gudekar: Student in MS Software Engineering at San Jose State University.

Sagar Bendale: Student in MS Software Engineering at San Jose State University.

Larkins Carvalho : Student in MS Software Engineering at San Jose State University. Contributed in

Nielet Dmello: Student in MS Software Engineering at San Jose State University.

7. Project details

GitHUB URL:

<https://github.com/omkargudekar/crime-analysis-engine>

Cloud URL:

<http://ec2-54-148-248-224.us-west-2.compute.amazonaws.com:8080/app/>

8. References:

[1] <https://developers.google.com/chart/>

[2] <https://www.data.gov/>

[3] <http://aws.amazon.com/documentation/>