SafeChicago

A platform that helps keep Chicagoans safe

CS 122 Project

Kei Irizawa | Adam Oppenheimer | Swayam Sinha | Shyamsunder Sriram

THE UNIVERSITY OF CHICAGO

Is Chicago one of America's most dangerous cities?

64018 cases of theft in 2017

Students Beaten, Cars Burned in Teen Attack on Halloween

Man shot during home invasion in East Garfield Park

29674 violent crimes in 2017

Headphones Stolen During Robbery On UChicago Campus

A teenage boy faces a robbery charge after police say he and one other person "forcibly took headphones" from a pedestrian on 56th Street.

By Tim Moran, Patch Staff | Oct 1, 2018 10:29 am ET

Is Chicago really unsafe or does it just have a bad rep?
How can we make it feel safer?

Problem 1: Biased perception of community areas

Suppose this box represents Chicago, and the boxes are community areas

MIXED	MIXED	SAFE
SAFE	SAFE	SAFE
MIXED	SAFE	MIXED
UNSAFE	MIXED	SAFE
UNSAFE	UNSAFE	MIXED

DISTORTED INFORMATION.

- Based on fear and emotion and not on data.
- Perception of safety is distorted. There are safe areas in communities in so-called “unsafe” areas, and there could be unsafe areas in ”safe” communities.

Problem 2: Situational data required to keep people safe…

The shortest route from Point A to Point B is not necessarily the *safest* route

if we think of entire Chicago street map
in terms of nodes and edges...

Solution we need…

Crime heatmap

MIXED	MIXED	SAFE
SAFE	SAFE	SAFE
MIXED	SAFE	MIXED
UNSAFE	MIXED	SAFE
UNSAFE	UNSAFE	MIXED

STORED DATA

(stored in dataset periodically updated)

Direction algorithm

DYNAMIC DATA

(requires real-time data/user input)

Project Overview

Our team

Kei Irizawa

Adam Oppenheimer

Swayam Sinha

Shyamsunder Sriram

Our platform goals

1 Birds-eye-view of safety across Chicago

- Provides visual representation on how safe an area as from a community and street point of view
- Provides insightful crime statistics by community area

2 Provide walking GPS route based on safety scores

- Create optimal safe walking route between point A and B
- Route based on safety scores assigned to each block

Data sources and libraries used

Data Sources

CHICAGO DATA PORTAL — CRIME DATA

OpenStreetMap — STREET NETWORK DATA

Google Maps API — LOCATION AND DIRECTION DATA

OpenWeatherMap — WEATHER DATA

Libraries used

Library	Function
OSMNX	Uses Open Street Maps, Geopandas, networkX and executes Dijkstra's algorithm
NetworkX	Store street data in Graph structure with intersection as node
Pandas	Data cleaning and aggregation
Matplotlib.pyplot	Plot density plots
Folium	HTML visual map with crime statistics
Dominate	Ability to code HTML websites in Python
Socrata	Pull recent data from Chicago data portal

Data sources overview

STORED DATA

Crime data

CHICAGO DATA PORTAL

Stored crime data based on community since 2017

Street map data

OpenStreetMap

Stored crime data based on community since 2017

DYNAMIC DATA

User input

- Starting destination
- Ending destination
- Safety threshold

Real time weather data

OpenWeatherMap

Since weather affects probability of crime happening, we extracted real time weather data

Location and direction data

Google Maps API

Gets data dynamically based on user input

Code overview

Stored data approach using pandas dataframes

CHICAGO DATA PORTAL

RAW DATA

Crime specific data

Each ✖ represents a crime that contains the following information

- Type of crime
- Location
- Date of crime

PANDAS OPERATIONS

Node specific data

Each D is a node that contains the following information:

- List of crimes
- Location

How do we weigh each crime differently?

Initial data problem

CRIME SPECIFIC

Each crime weighed the same here…

NODE SPECIFIC

Safety of each node determined by number of crimes

Assigning crime weights

Data we need

Crimes weighed by probability of harm to e.g. (public indecency vs. murder)

Safety of each node determined by aggregate weights of crimes

Assigned crime weights

```python
def get_crime_weights():
    crime_weights = {'THEFT': 1, 'BATTERY': 2, 'CRIMINAL DAMAGE': 0.5, 'ASSAULT': 2,
                     'DECEPTIVE PRACTICE': 0.1, 'OTHER OFFENSE': 0.1, 'NARCOTICS': 0.1,
                     'BURGLARY': 1, 'ROBBERY': 1, 'MOTOR VEHICLE THEFT': 0.1,
                     'CRIMINAL TRESPASS': 0.1, 'WEAPONS VIOLATION': 0.5,
                     'OFFENSE INVOLVING CHILDREN': 0.5, 'CRIM SEXUAL ASSAULT': 1,
                     'PUBLIC PEACE VIOLATION': 0.1, 'INTERFERENCE WITH PUBLIC OFFICER': 0.1,
                     'SEX OFFENSE': 0.1, 'PROSTITUTION': 0.1, 'HOMICIDE': 20, 'ARSON': 2,
                     'LIQUOR LAW VIOLATION': 0.5, 'STALKING': 0.5, 'GAMBLING': 0.1, 'KIDNAPPING': 10,
                     'INTIMIDATION': 0.5, 'CONCEALED CARRY LICENSE VIOLATION': 0.5,
                     'OBSCENITY': 0.5, 'NON-CRIMINAL': 0, 'HUMAN TRAFFICKING': 10,
                     'PUBLIC INDECENCY': 0.5, 'OTHER NARCOTIC VIOLATION': 0.1,
                     'NON-CRIMINAL (SUBJECT SPECIFIED)': 0}
    return crime_weights
```

Note: we used our intuition to assign these crime weights but further research is recommended.

Safety can be determined by variety of factors

Weather

Does weather influence crime? The data say yes - Chicago Tribune
https://www.chicagotribune.com/news/data/ct-crime-heat-analysis-htmlstory.html
Aug 23, 2017 - Correlating those day-by-day **crime** records with changes in average daily **temperature** shows some types of **crime** are much more likely to increase when the **weather** is hotter and decrease when cooler. ... However, homicide does not show a **relationship** to **temperature**, nor do drug **crimes**.
You visited this page on 3/8/19.

A Rise in Murder? Let's Talk About the Weather - The New York Times
https://www.nytimes.com/2018/09/21/.../a-rise-in-murder-lets-talk-about-the-weather.ht...
Sep 21, 2018 - The **correlation** between heat and **crime** suggests the need for more ... **weather** can't really explain big increases in **murder** in **Chicago** or ...

*Online research

Season

*Generated from our crime data

Time of day

*Generated from our crime data

MAIN INSIGHTS

Extreme temperatures decrease probability of crime

Hot weather holidays make summer months a hot time for crime (- pun intended)

Significant drop in crime during the early morning

Divided safety factors into categorical variables

Weather

No. of crimes in Chicago, by temperature (online research)

Season

Time of day

CATEGORICAL VARIABLES USED

Extreme temperature (<10F and >95F) Vs. non extreme temperature

Same season vs. Different season

Early morning vs. Other times of day

Assigned a ”safety score” for each node

=

Crime weight

Aggregate crime weights by node

✖

Profile weight

Greater weight given to crimes with same season and/or time

✖

Weather Weight

Less weight given to extreme temperatures

✖

Time decay

greater weight given to present data

Aggregate crime weights by node

1 x if season and time categorical variables **are the same**

1/5 x if **one** of the categorical variables **are different**

1/25 x if **both** of the categorical variables **are different.**

1 x if <u>not</u> an extreme temperature

3/4 x if extreme temperature

0.998^t where t is the number of days since the crime

Computing average safety score for each node

Profile categorical combinations (Season and Time)

Time	Early morning vs. Other times
Seasons	Spring Summer Fall Winter

8 different possible assumptions

ASSUMPTIONS ARE UNIVERSAL

Each node has the same time and season (because they are all in the same city)

CAN COMPARE SCORES FOR EACH NODE

Heatmap program overview

CRIME CIRCLES

Uses **crime specific data** to **generate folium circles** at each crime location.

Increasing weight of crime

ZOOM OUT

CITY

COMMUNITY

STREET

ZOOM IN

STATISTICS

Uses **crime specific data** to **generate statistics** for each community area. Includes:

- Histogram of safety scores
- Crime counts by time of day, month and season

AVERAGE SAFETY SCORE

Uses **node specific data** to generate safety score for most dangerous node for each community area. Can use this for comparison of overall safety for each community

Heatmap program overview

CRIME CIRCLES

Uses **crime specific data** to **generate folium circles** at each crime location.

Increasing weight of crime

ZOOM OUT

CITY

COMMUNITY

STREET

ZOOM IN

STATISTICS

Uses **crime specific data** to **generate statistics** for each community area. Includes:

- Histogram of safety scores
- Crime counts by time of day, month and season

AVERAGE SAFETY SCORE

Uses **node specific data** to generate safety score for most dangerous node for each community area. Can use this for comparison of overall safety for each community

Heatmap program use cases

City view

Can compare crime statistics across community areas

Community view

Understand safe and unsafe areas in a community

Street view

Determine whether a crime has happened near a specific location

Transition into dynamic data

Revisiting direction algorithm

DYNAMIC DATA

(requires real-time data/user input)

REQUIRED USER INPUT/DYNAMICAL DATA

- Start and end point
- Safety threshold
- Weather
- Time of day

Requires situational safety score (one of our 8 assumptions)

How to determine correct route with safety threshold?

Threshold vs. direction

For some users that are not risk averse
(based on safety threshold input)
A to B might be a viable route…

OUR SOLUTION

Increase **length of node** based on safety threshold so that *Dijkstra algorithm* can avoid certain routes

Factor to increase node length by based on safety score percentile (higher is more dangerous)

Direction algorithm implementation

1

Gather user input

- Starting destination
- Ending destination
- Safety threshold

2

Google Maps Geocode API

Convert start and end destination to Location information (latitude and longitude)

3

Apply Dijkstra's Algorithm with nearest node

Use user safety threshold input to modify lengths of streets and apply Dijkstra's algorithm.

4

Open Street Map

Use latitude and longitude information to get nearest node to starting destination. From here we collect nodes based on Dijkstra's Algorithm

5

Google maps Direction API

Used to get step-by-step direction based on travelling from node to node

Demonstration