FINDING THE SHORTEST PATH PREVENTING SEXUAL HARASSMENT THROUGH ALGORITHMS

Juan Felipe Restrepo Buitrago
Universidad Eafit
Colombia
jfrestrepb@eafit.edu.co

Sara Valentina Cortes Manrique
Universidad Eafit
Colombia
svcortesm@eafit.edu.co

Andrea Serna
Universidad Eafit
Colombia
asernac1@eafit.edu.co

Mauricio Toro
Universidad Eafit
Colombia
mtorobe@eafit.edu.co

ABSTRACT

Sexual harassment is a daily concern for Medellin's women. Thanks to a Medellin's mayoralty which made a survey to 1000 women, we know that 85% of them have suffered sexual harassment [9]. We have to take control of this situation, due to the insecure feeling present in the majority of women, so they could live more calm and obviously comfortable going to anywhere. With this project we hope to create a solution to this problem using an algorithm to find a safer path which takes the least time posible to get from one location to another. Trying to avoid sexual harassment on the streets is imposible without considering crime, which helps us both, men and women, to feel less comfortable out in the city.

Key words

Shortest route, street sexual harassment, identification of safe routes, crime prevention

1. INTRODUCTION

As we said lately, people don't feel safe in the city and are worried, especially women. If the city were safer, surely our parents hadn't taught us to be extremely cautious on the streets as the majority probably are. According to the given data by The Legal Medicine Nacional Institute in the first trimester of 2022 we had 6.336 violent homicides, 848 more than the first quarter of 2021. Medellin is in fact one of the most affected cities of the country by this matter with around 232 homicides [14]. Despite the fact that the number of cases in Medellin have reduced, people are still worried mostly for going out at night with a 35% of safeness according to what Medellin's people think about crime in the last 3 years [8]. We want people to feel safer and more commfortable while going on the street.

1.1. The problem

The problem we're trying to solve is to find three paths to lead people to their destiny. One of them will be the shortest without having in mind the danger of the path is going through, other one will be the safest without having in mind the distance and the last one will consider proportionally both distance and safety. These three paths are important because of the situation the person using the algorithm is living through. In first place, if they need to hurry and get as quickly as possible to a place, they will probably avoid safety parameter, however on the other hand, someone could probably not need time, but safety or another person could need both parameters in their trip.

1.3 Structure of the article

Next, in Section 2, we present work related to the problem. Then, in Section 3, we present the datasets and methods used in this research. In Section 4, we present the algorithm design. Then, in Section 5, we present the results. Finally, in Section 6, we discuss the results and propose some directions for future work.

2. RELATED WORK

Below, we explain four works related to finding ways to prevent street sexual harassment and crime in general.

2.1 Safetipin: A Free Map-Based Application, Which Helps Users to Which Areas They Would Like to Pass Through and Which Ones to Avoid.

Safetipin is a mobile application which allows people to check whether a location is safe or not, and it finds the best paths to take people to a location avoiding unsafety places where dangers such as crime and street sexual harassment could occur. Although the safety of somewhere is mainly calculated by the Safetipin's team, users can give their opinions of a place, improving the score's accuracy of those places. Besides, people can tell the app which places they want to avoid. Safetipin was created by Kalpana Viswanath to treat the safety problem in Delhi [12].

Talking in algorithm terms, they are based on GIS to collect their information and their application runs machine learning [6].

2.2 The Safe Route: Multi-Options Route Finder for Cyclists.

The Safe Route is an application developed especially for bike drivers. This app provides the user multiple options of paths he could go through from the fastest to the largest and safest in terms of traffic and accidents. The Safe route was created by the company Futurice. The problem this app is considering is road insecurity for cyclists in Sweden. They hope to encourage people to ride a bike, giving them safe ways and helping the environment [10].

The parameters this app considers for providing a navigation based on safety are traffic jams, road work, crossings, poor surfaces, weather and accident statistics [16] Data about the kind of algorithm this app uses is not given.

2.3 TomTom: Algorithms Prioritizing Safety Over Speed

TomTom is an application, which finds routes to take people form one location to another. Lately this app is taking into account the safety of some roads above others. This new function was implemented because of the danger drivers were exposed to, one study was made in Finland that the fastest route to Koli National Park, was the most dangerous mainly on winter, because of the snow. So now this app takes parameters such as weather, quality of the road, etc. for finding a path [1].

TomTom's engine is based on the A* algorithm [6].

2.4 Path Community: Red Flags on the Streets

Path is an application that suggests routes in which the user is not likely to be involved into a harassment, assault or attack situation. It was created by Harry Mead. This is a user's opinion-based application, where a user can highlight a dangerous area as red flags, which helps the app find the safest route to one point to another by avoiding these red flags [11].

3. MATERIALS AND METHODS

In this section, we explain how the data were collected and processed, and then different alternative path algorithms that reduce both the distance and the risk of sexual street harassment.

3.1 Data collection and processing

The map of Medellín was obtained from *Open Street Maps* (OSM)[1] and downloaded using the Python API[2] OSMnx. The map includes (1) the length of each segment, in meters; (2) the indication of whether the segment is one-way or not, and (3) the known binary representations of the geometries obtained from the metadata provided by OSM.

For this project, a linear combination (LC) was calculated that captures the maximum variance between (i) the fraction of households that feel insecure and (ii) the fraction of households with incomes below one minimum wage. These data were obtained from the 2017 Medellín quality of life survey. The CL was normalized, using the maximum and minimum, to obtain values between 0 and 1. The CL was obtained using principal components analysis. The risk of harassment is defined as one minus the normalized CL. Figure 1 presents the calculated risk of bullying. The map is available on GitHub[3] .

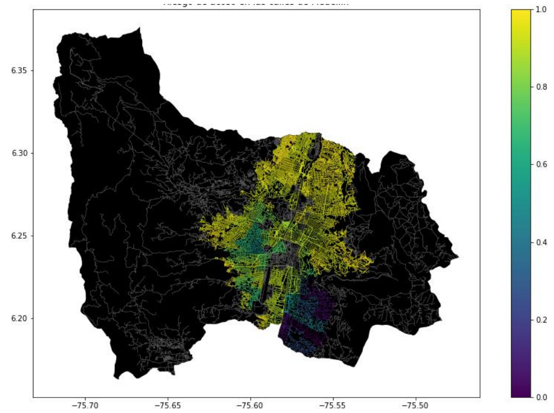


Figure 1. Risk of sexual harassment calculated as a linear combination of the fraction of households that feel unsafe and the fraction of households with income below one minimum wage, obtained from the 2017 Medellín Quality of Life Survey.

3.2 Algorithmic alternatives that reduce the risk of sexual street harassment and distance

In the following, we present different algorithms used for a path that reduces both street sexual harassment and distance.

3.2.1 Breadth-First Search

Is an important graph search algorithm that is useful for analyzing and solving graph problems.

Breadth First Search starts by searching start node, followed by its adjacents nodes, then all nodes can be reached by a path from the start node containing two edges, three edges and so on. Normally, BFS visits all vertices in a graph *G* that are *k* edges away from the source vertex s before visiting any vertex *k + 1* edges away. This is done until no more vertices are reachable from *s* [3].

[1] https://www.openstreetmap.org/

[2] https://osmnx.readthedocs.io/

[3]https://github.com/mauriciotoro/ST0245Eafit/tree/master/proyecto/Datasets

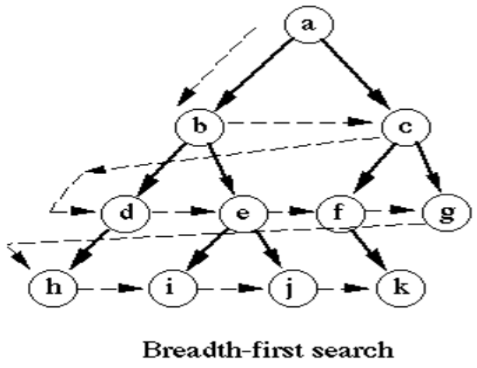


Figure 1 Example of Breadth First Search Algorithm [17]

3.2.2 Depth First Search

Is and algorithm for searching a graph or three data structure. The algorithm starts at the root (top) node of a tree and goes as far as it can down a given branch (path), then backtracks until it finds an unexplored path, and then explores it. The algorithm does this until the entire graph has been explored [4].

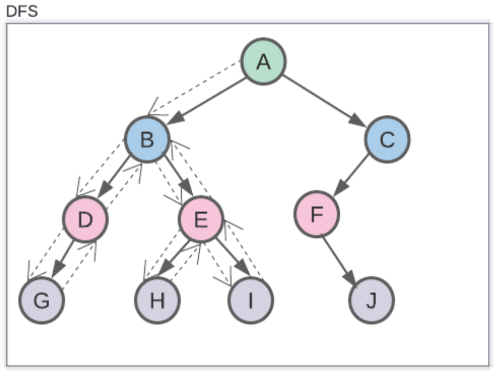


Figure 2DFS Representation example [7]

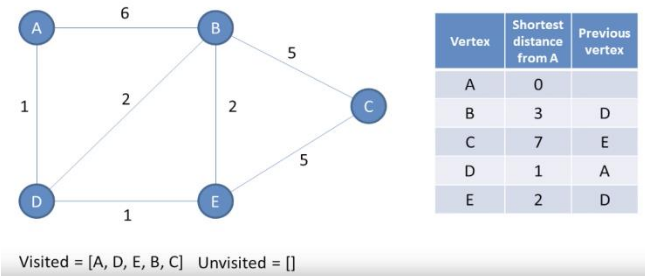
3.2.3 Dijkstra's Shortest Path Algorithm

It is an algorithm to find the shortest path from a starting node to a target node in a weighted graph. Dijkstra algorithm creates a tree of shortest paths from the starting vertex, the source, to all other points in the graph.

Dijkstra’s algorithm, published in 1959 and named after its creator Dutch computer scientist Edsger Dijkstra, can be applied on a weighted graph. The graph can either be directed or undirected. One stipulation to using the algorithm is that the graph needs to have a nonnegative weight on every edge [5].

Figure 3 The result of Dijkstra’s algorithm starting at node v is a shortest path tree rooted at v, such that the path from root v to any other node in the tree is the shortest path distance. The shortest path tree is a spanning tree, meaning that it is a subgraph which includes all the vertices of G [13].

3.2.4 Bellman-Ford Algorithm

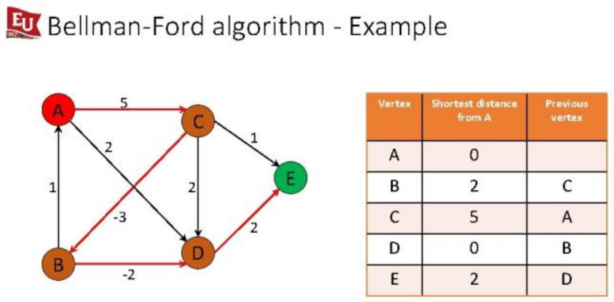


Vertex	Shortest distance from A	Previous vertex
A	0	
B	3	D
C	7	E
D	1	A
E	2	D

Visited = [A, D, E, B, C] Unvisited = []

The **Bellman-Ford algorithm** is a graph search algorithm that finds the shortest path between a given source vertex and all other vertices in the graph. This algorithm can be used on both weighted and unweighted graphs.

Like Dijkstra's shortest path algorithm, the Bellman-Ford is guaranteed to find the shortest path in a graph. Though it is slower than Dijkstra's algorithm, Bellman-Ford is capable of handling graphs that contain negative edge weights, so it is more versatile. It is worth noting that if there exists a negative cycle in the graph, then there is no shortest path [2].


Bellman-Ford algorithm - Example

Vertex	Shortest distance from A	Previous vertex
A	0	
B	2	C
C	5	A
D	0	B
E	2	D

Figure 4Bellman-Ford algorithm example [15]

ACKNOWLEDGEMENTS

The authors thank Professor Juan Carlos Duque, Universidad EAFIT, for providing the data from the 2017 Medellín Quality of Life Survey, processed in a *Shapefile*.

REFERENCES

1. Beedham, M. (2022, January 17). *How Routing Algorithms Can Prioritize Safety Over Speed*. Retrieved from TomTom:

https://www.tomtom.com/blog/navigation/map-data-to-suggest-the-safest-route/

2. Brilliant. (2022, August 13). *Bellman-Ford Algorithm*. Retrieved from Brilliant.org: https://brilliant.org/wiki/bellman-ford-algorithm/

3. Brilliant. (2022, August 13). *Breadth-First Search (BFS)*. Retrieved from Brilliant.org: https://brilliant.org/wiki/breadth-first-search-bfs/

4. Brilliant. (2022, August 13). *Depth-First Search (DFS)*. Retrieved from Brilliant.org: https://brilliant.org/wiki/depth-first-search-dfs/

5. Brilliant. (2022, August 13). *Dijkstra's Shortest Path Algorithm*. Retrieved from Brilliant.org: https://brilliant.org/wiki/dijkstras-short-path-finder/

6. Buve, R. (n.d.). *What routing algorithms are most likely being used by navigation services like Garmin and TomTom? What factors are included in the final algorithm?* Retrieved from Quora: https://www.quora.com/What-routing-algorithms-are-most-likely-being-used-by-navigation-services-like-Garmin-and-TomTom-What-factors-are-included-in-the-final-algorithm

7. Cortes, S. (2022, August 14). *Lucidspark*. Retrieved from lucid.app: https://lucid.app/lucidspark/bcf35ea3-ce3d-42d2-b9d0-7b184688fb31/edit?invitationId=inv_7b61f247-0e94-4ff5-a133-cac609c1ad3f#

8. *Criminalidad en Medellín*. (2022, August). Retrieved from NUMBEO: https://es.numbeo.com/criminalidad/ciudad/Medell%C3%ADn

9. *El 90,1 por ciento de la mujeres no denuncia el acoso callejero*. (2019, April 29). Retrieved from El Tiempo: https://www.eltiempo.com/colombia/medellin/el-90-1-por-ciento-de-las-mujeres-no-denuncia-el-acoso-callejero-en-medellin-355056

10. Lauri, J. (2022, March 14). *The Safe Route - improving the safety of cyclists in Sweden*. Retrieved from Futurice: https://futurice.com/blog/if-you-had-a-choice-would-you-select-speed-over-safety

11. Mustafa, T. (2022, January 18). *New safety app helps you find the best route home*. Retrieved from Metro 50: https://metro.co.uk/2022/01/18/new-safety-app-helps-you-find-the-best-route-home-15932866/#metro-comments-container

12. Pareek, S. (2015, July 2). *An App that Helps You Decide the Safest Route Home*. Retrieved from The Better India: https://www.thebetterindia.com/22908/app-helps-decide-safest-route-home/

13. Rkpandey. (2022, August 14). *Dijkstra's Algorithm for Shortest Path*. Retrieved from rkpandey.com: http://rkpandey.com/AlgorithmHelper/shortest/path/2017/05/16/dijkstra.html

14. *Se registraron 6.336 muertes violentas en Colombia en el primer trimestre de 2022*. (2022, May 5). Retrieved from infobae: https://www.infobae.com/america/colombia/2022/05/05/se-registraron-6336-muertes-violentas-en-colombia-en-el-primer-trimestre-de-2022/

15. Slidetodoc. (2022, August 14). *BellmanFord Algorithm CSCI 385 Data Structures Analysis of*. Retrieved from slidetodoc.com: https://slidetodoc.com/bellmanford-algorithm-csci-385-data-structures-analysis-of/

16. *The Safe Route*. (n.d.). Retrieved from Futurice: https://futurice.com/saferoute

17. Vivadifferences. (2022, August 14). *8 Difference Between DFS (Depth First Search) And BFS (Breadth First Search) In Artificial Intelligence*. Retrieved from vivadifferences.com: https://vivadifferences.com/difference-between-dfs-and-bfs-in-artificial-intelligence/