Safety-oriented Apartment Listing Site for Students

Manqiu Liu
mliu437@gatech.edu

Chukang Zhong
czhong38@gatech.edu

Yucheng Liang
yliang350@gatech.edu

Qian Huang
qhuang301@gatech.edu

Jia Shi
jshi351@gatech.edu

1 INTRODUCTION

College students nowadays are facing increasing exposure to urban crimes. Various measures have been instituted by both campus and governmental authorities in recent years to enforce students' safety. Our team believes that amidst general success, however, there remains a blind spot. The safety measure for students living off-campus largely depends on their living situation and its surroundings, but none of the existing apartment listing sites such as Apartments.com offer truly robust functionalities in this regard. We attempt to take into account the fact that people's safety is affected by their daily activities in the city as they travel to campuses, workplaces, local groceries, pharmacies, gyms and other entities.

2 PROBLEM DEFINITION

The goal of our project is to create a more dynamically safety-oriented apartment listing site that considers each property listing not as a fixed location, but as a web of routes centered at the apartment building and expanding into the wider neighborhood. Put formally, for each apartment building within a user-defined radius, our application will compute a safety score by combining the safety score of its location and of all the routes between this building and user's most frequented locations.

3 SURVEY

When looking for apartments, there are a wide variety of evaluation criteria. According to a literature survey conducted in 2017, however, safety issues specific to apartment buildings are not treated as a top priority in past research and cannot be found on popular apartment listing sites [2]. Below we will discuss several common practices of evaluating the safety of locations and routes.

To evaluate the safety of a location, a classical method is hotspot analysis, which detects areas with high crime density [12]. This analysis has its limitation in that it is completely based on historical data, thus, people have developed predictive models [3] that take in additional temporal and environmental factors [14] as well as introduced deep learning [8] [13]. Since these methods usually require large amount of data and high computing costs, we will modify the classic hotspot analysis with advanced ideas, and take a compromise solution.

AbuSalim's research compares two popular shortest path algorithms [1]. To evaluate the safety of a route, Kaur's Safe Routing system combines crime data into navigation but is prone to misleading results because not every kind of crime has an equal effect on one's safety [9]. Nenadić's MCDM model improves by combining five quantitative criteria and two qualitative criteria [11]. Galbrun's biobjective model explores similar ideas [4].

Airbnb search ranking models are an exemplar of industry-level practices at present. To facilitate real-time personalization for both customers and property owners, researchers have developed two embedding approaches [10] for each type of user. Combined with domain knowledge, they added booked listings as a global context in a skip-gram model [5]. They have also applied neural networks [6] [7]. However, these methods require millions of click sessions to train the embeddings and are not applicable to our project.

4 INNOVATION

We have the following four innovations in our project:

1 We designed a safety-oriented apartment recommendation site for students and incorporated the safeness of apartment's surrounding areas and the safeness of their daily routes.

2 We encoded geographical data into grids and built predictive model to evaluate the safeness of each grid,

which incorporated the adjacent effects of crime incidents and public facilities as well as reduced computational cost.

3 We visualized the safeness of areas and routes as well as the total safety score, which explicitly showed the overall safeness of chosen student apartments.

4 We enabled our users customize their daily visit spots and calculate safety score accordingly.

5 METHOD

5.1 Data Preparation

5.1.1 Apartment Data. We obtained our apartment data from Apartments, which is a website providing detailed information about apartments, including name, price, location, etc. As we are supposed to focus on the Atlanta area, we only collect apartments around Atlanta city. Therefore, we could provide a basic view of apartments when the user first opens the website. In this part, we mostly use crawler to pull the webpage data from the website and then parse them with BeautifulSoup. Additionally, we clean the data and store them locally to shorten the delay of user request.

5.1.2 Crime Data. We obtained crime data from the Atlanta Police Department website, and the dataset contains serious crimes reported to law enforcement agencies in the Uniform Crime Reporting program, which contains murder, forcible rape, robbery, etc. The data for each crime is with latitude and longitude, so we could generate our crime map based on it.

5.1.3 Geographic Data. We collected our geographical data from OpenStreetMap, which contains information of geomorphic characteristics, public transportation, public facilities, etc. According to our literature review, public utilities could have positive influence on the safeness of surrounding areas. Thus we have picked the locations of public transportation and public facilities such as police departments, government offices, hospitals, churches, schools, etc.

5.2 Location Safety Score Evaluation

5.2.1 Grid Encoding. To analyze crime incidents as location features and to reduce real-time computation cost, we split the area of Atlanta into 260*280 grids based on longitude and latitude. The we counted the number of crime incidents, public facilities in each, and stored the numbers in a multidimensional matrix with a shape of 260*280*D*T, where D stood for number of features, with each feature being stored in each layer, and T stood for time of features. In order to incorporate adjacent crime incidents for each grid, we applied kernel method on the matrix as the Figure 1 shows.

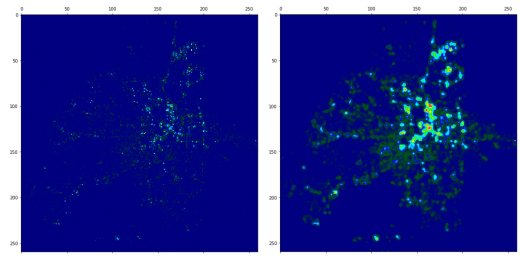

Figure 1: Original and kernelized crime data grids

5.2.2 Time Series Prediction. Based on the obtained grid data, we split them and got time series data for each location. Then, we tried to decompose time series and utilize additive model with fitting non-linear trends with seasonality, to predict the sum of crimes for the area in next time unit.

5.2.3 Geographical Features Prediction. With the geographical features matrix as independent variable and crime data matrix as dependent variable, we were able to perform regression on the panel data to model the public facilities' impact on location safeness. As the public facilities can have influence on adjacent areas, we applied kernel method on our geographical features matrix to simulate this effect. We built multivariate linear regression model and got the parameters of each features, which were then used in our prediction to the future crime incidents.

5.3 Route Safety Score Evaluation

5.3.1 Route Planning. We have two parts of route planning. First, we use Google Map API [15] to provide a basic route between two locations, which can be default or user defined. We accept multiple transport modes because some students prefer to walk or ride while others prefer to drive. With our route safety evaluation model, we can score each route based on its distance, safety, and convenience. Finally, we can re-plan the route by avoiding the dangerous crime spot along with the route and therefore give a "green route" for the user. This

could provide excellent user experience because it can not only give score for each route to evaluate the safety of the apartment, but also give a recommendation if the user chooses to live in a specific apartment.

5.3.2 Route Safety Evaluation. The decoded polyline using Google Maps Directions API's polyline shows all the waypoints between two locations. Therefore, we compute the safety score of a route and evaluated the safety status by utilizing the waypoints data. For each waypoint, it should fall into a certain grid of the map matrix. Since every grid is assigned with a safety score, we can obtain the safety score of a certain waypoint by taking the safety score of the grid where the waypoint falls. To compute the safety score of a route, we simply add the safety score of all the waypoints on the route. As a result, we obtained the route safety score. Currently, we didn't take the length of the route into account. For further analysis, we will compare the results of using safety score without normalizing the route length and using safety score with normalization to see which one is more effective.

5.4 Visualization

5.4.1 Map with Safety Evaluation & Recommendation List. Our visualization is largely map-based. For this reason, we use Google Maps APIs as our primary source of visualization tools. Using Google Maps JavaScript API, our basic map includes features such as zooming and dragging. Each apartment listing is represented as a marker on the map, with green markers corresponding to the top 50 apartments in our recommendation list, as the Figure 2 shows.

On top of the basic map, we add a Heatmap Layer based on our crime prediction. Using our geographical and time-series predictions, we use heatmap to create an intuitive visualization of the relative safety of any given area.

On the right side of the page, there is a real-time update recommendation list based on the user-defined destination in the drop-down menu above. The safety scores displayed are calculated based on the combinations of the basic score for each apartment and route scores for closest destinations of each type.

5.4.2 Route with Safety Evaluation. When an apartment is selected by the user, we will visualize all the

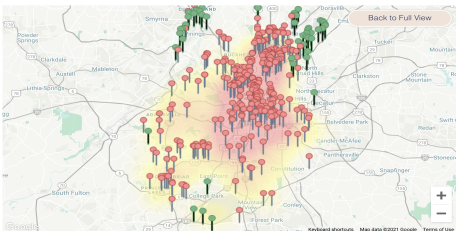

Figure 2: Heatmap and apartment layer overview

routes between the apartment and the closest user-defined destinations. Each route has a safety score based on our route safety evaluation. In order to help users to gain intuitive insights on the safety status of the routes, we assign color scale from green to red corresponding to routes with higher safety score to lower safety score as the Figure 3 shows. In addition, the color gradient and distribution of the routes will change dynamically with users changing the destinations. The specific score value, distance and name of destinations are displayed on the right side of the map.

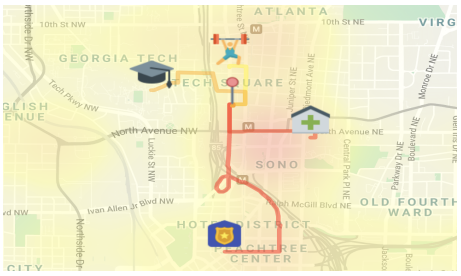

Figure 3: Route planning and safety evaluation

5.4.3 Crime Data Analytics. We analyzed crime data for Atlanta in both temporal and spatial dimensions, and visualized it with Plotly JavaScript Graphing Library. We hope this page will give our users a broader view of local crime in Atlanta and help them better choose apartments.

5.4.4 Dashboard for Apartment Comparison. We also provided a dashboard to help our users to more directly compare the scores of their favorite apartments. In the control panel, they can conveniently select the frequently visited facilities to recalculate the total score of the apartment. This feature targets scenarios to help users with the final apartment comparison and selection. In addition, if the user is logged into our system with a Google account, the apartments they have saved

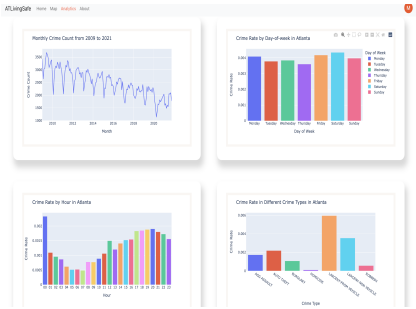

Figure 4: Crime Data Overview

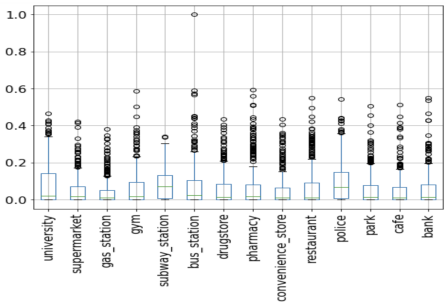

Figure 6: Safety score boxplot

while browsing our map will be stored in our server database and will be displayed when the users open the dashboard under their account.

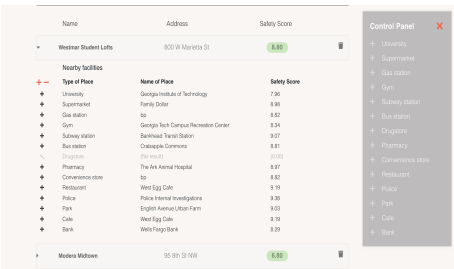

Figure 5: Dashboard

6 EXPERIMENTS

6.1 UI Experiment

In route visualization, we assign green colors to routes with higher safety scores and yellow and red colors to routes with lower safety scores to enable users acquire insights about the safety status of the routes. However, we noticed that the color distinction between different routes is subtle and not easy to distinguish. The reason is that the safety scores of each route follow a left-skewed distribution, which leads to most of the route's color being green.

We then draw a boxplot to visualize the score distribution. Most of the scores are distributed between 0.02 and 0.08.

To determine the best color scale for distinguishing routes of different safety scores, we did an experiment and tested different safety score domains for our color scales. We tried [0,0.2], [0,0.4], [0,0.6], [0,0.8] as the domain of safety score corresponding to color scale ["green", "yellow", "orange", "red"], and concluded that [0,0.4] is the most effective domain for distinguishing the routes of different safety score.

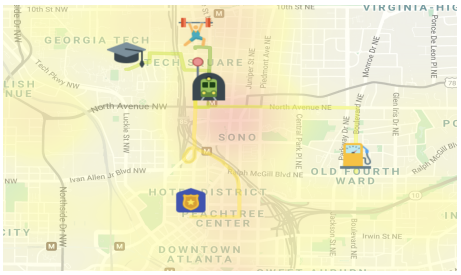

Figure 7: Route visualization with safety score domain [0,1]

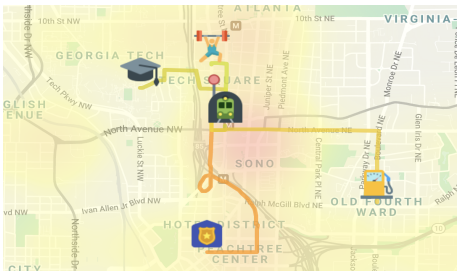

Figure 8: Route visualization with safety score domain [0,0.4]

6.2 Heatmap Color Scale Test

We also test different color scales for our heatmap to improve user experience. We first set our heatmap color scale as ["green", "yellow", "orange", "red"]. The red area on map indicates that the safety score of that area is low. If there is no crime in this area, then there is no color for this area. We noticed that this kind of color scale may bring misleading information to users, since

most people think that the green areas are the safest area. In fact, areas with no color are the safest areas.

To improve user experience and let users get clearer insights about safety status of different areas through our visualization, we change the heatmap color scale to ["yellow", "orange", "red"]. In this way, the colored area which are yellow area and red area are relatively more dangerous than the area with no color.

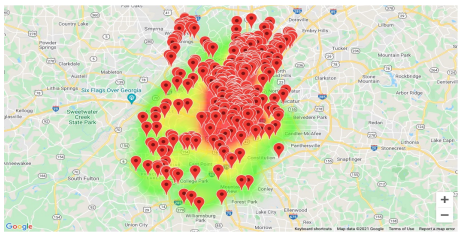

Figure 9: Previous Heatmap

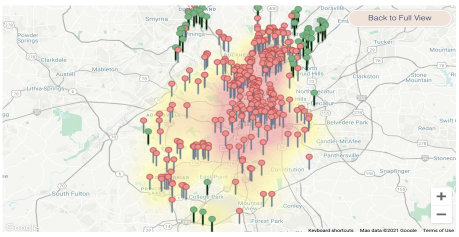

Figure 10: Heatmap with New Color Scale

6.3 Recommendation Sensitivity Analysis

According to our recommendation, we set the color of the top 50 apartments with highest safety score to green on the map and the color of remaining apartments to red. We then do sensitivity analysis on our apartment recommendation algorithm. We changed different combinations of destinations and saw whether the recommendation results had changed with the change of destinations. We tested 30+ combinations and concluded that if we select different destinations, the recommendation results will change, which shows that our recommendation algorithm is sensitive to users personalized needs.

The following picture shows that as we increase the number of destinations, more apartments located in Midtown and Downtown will show up in our top 50 apartment recommendation list.

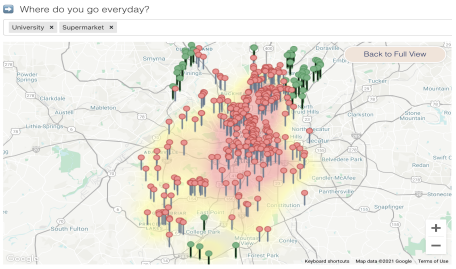

Figure 11: Recommendation with less destination choices

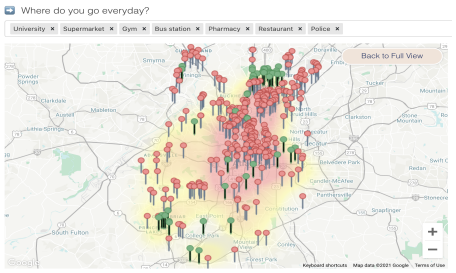

Figure 12: Recommendation with more destination choices

7 EVALUATION

7.1 Apartment Recommendation

Our heatmap and route safety ratings are based on our initial Atlanta safety projection. Our projection model was based on Atlanta crime data from the previous ten years, up to October 2021. As a result, we used data from November 2021 as a test set to see if our apartment recommendation accurately reflects the safety of each neighborhood.

We contrasted crime data from November with our apartment recommendation, which is solely based on location safety, as seen in the graph below. We can see that the November crime incidences had little overlap with our recommended apartments, indicating that our recommendation is sound and accurate.

7.2 Evaluation of Visualization and Product Design

We invited 17 Georgia Tech students to try our product and prepared a questionnaire to collect their input in order to see if it was easy to use and valuable to our users. Our survey inquired as to whether the product is simple to use, whether the visualization is clear, and

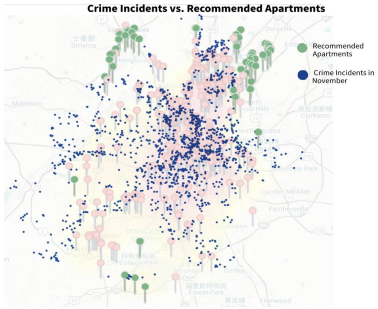

Figure 13: Crime Incidents v.s. Recommended Apartments

whether the analytics are useful. We also gathered feedback from our users and calculated our product's NPS. The following is how we presented our findings:

We discovered that our users are generally satisfied with our product, with 64% NPS score. The route planning and evaluation element is the most popular of all the features.

Figure 14: User Survey

We also gathered our consumers' future expectations for our product, and based on that feedback, we'll be adding suitable illustrations to our visualizations to make our product more intelligible, as well as altering our information display to avoid having too much information on the same page.

8 CONCLUSION AND DISCUSSION

We created an interactive apartment recommendation service to help college students locate safe off-campus housing. This site uses React.js for the frontend, Node.js for the backend, and MongoDB for the database.

We assessed users' everyday travels from the apartment to areas like universities, stores, and other places while evaluating the safety of a certain unit. Then We constructed a predictive model based on apartment data from online scraping and crime statistics from the previous ten years to calculate a safety score for each grid on the map, then mapped apartments and routes to each grid to arrive at our safety score.

We used heatmap layers to disclose the safety of corresponding sites, and each route was given a color to represent its level of safety. to Our users may immediately interact with our map and apartment recommendation lists to gain a thorough description of the safety evaluation, save and compare apartments in the dashboard, and learn more about Atlanta's overall safety statistics on our analytics page.

Our user survey yielded a lot of good responses. Based on our findings and comments from users, we plan to improve the safety score methodology and related visualizations in the future. Currently, our safety score does not take into account the variation in safety over time. Because our analytics demonstrate that the safeness at different times of day and night differs, we planned to include the safety evaluation in our subsequent safeness evaluation algorithm.

8.1 Plan of Activities

Sept. 08 - Oct. 15:

- Dataset search + idea generation - *Everyone*
- **Project proposal** - *Everyone*

Oct. 16 - Nov 7:

- Data scraping + cleaning - *Yucheng, Manqiu*
- Crime data prediction - *Chukang, Manqiu*
- Crime data visualization - *Qian, Chukang*
- Routes planning + visulization - *Jia, Yucheng*
- Metrics defining and computation - *Qian, Jia*
- **Progress Report** - *Everyone*

Nov 7 - Dec 2:

- Ranking Construction - *Chukang, Manqiu*
- Website building - *Qian, Yucheng, Jia*
- Experimentation - *Everyone*
- functionality testing - *Everyone*
- **Final Report/Poster/Video** - *Everyone*

REFERENCES

[1] Samah WG AbuSalim, Rosziati Ibrahim, Mohd Zainuri Saringat, Sapiee Jamel, and Jahari Abdul Wahab. 2020. Comparative analysis between dijkstra and bellman-ford algorithms in shortest path optimization. In *IOP Conference Series: Materials Science and Engineering*, Vol. 917. IOP Publishing, 012077.

[2] Ivana Brkanić. 2017. Housing quality assessment criteria. *Electronic Journal of the Faculty of Civil Engineering Osijek-e-GFOS* 8, 14 (2017), 37–47.

[3] Adelbert W Choi, Vio Jianu C Mojica, Frumencio F Co, and Robert Neil F Leong. 2016. Spatial Regression Analysis of Violent Crimes in the National Capital Region. (2016).

[4] Esther Galbrun, Konstantinos Pelechrinis, and Evimaria Terzi. 2016. Urban navigation beyond shortest route: The case of safe paths. *Information Systems* 57 (2016), 160–171.

[5] Mihajlo Grbovic and Haibin Cheng. 2018. Real-time personalization using embeddings for search ranking at airbnb. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 311–320.

[6] Malay Haldar, Mustafa Abdool, Prashant Ramanathan, Tao Xu, Shulin Yang, Huizhong Duan, Qing Zhang, Nick Barrow-Williams, Bradley C Turnbull, Brendan M Collins, et al. 2019. Applying deep learning to airbnb search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1927–1935.

[7] Malay Haldar, Prashant Ramanathan, Tyler Sax, Mustafa Abdool, Lanbo Zhang, Aamir Mansawala, Shulin Yang, Bradley Turnbull, and Junshuo Liao. 2020. Improving deep learning for airbnb search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2822–2830.

[8] Hyeon-Woo Kang and Hang-Bong Kang. 2017. Prediction of crime occurrence from multi-modal data using deep learning. *PloS one* 12, 4 (2017), e0176244.

[9] Allan Mariano de Souza, Torsten Braun, Leonardo C Botega, Raquel Cabral, Islene C Garcia, and Leandro A Villas. 2019. Better safe than sorry: a vehicular traffic re-routing based on traffic conditions and public safety issue. *Journal of internet services and applications* 10, 1 (2019).

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[11] Dragana Nenadić. 2019. Ranking dangerous sections of the road using MCDM model. *Decision Making: Applications in Management and Engineering* 2, 1 (2019), 115–131.

[12] Jerry H Ratcliffe and Michael J McCullagh. 1998. Aoristic crime analysis. *International Journal of Geographical Information Science* 12, 7 (1998), 751–764.

[13] Jiao Sun, Mingxuan Yue, Zongyu Lin, Xiaochen Yang, Luciano Nocera, Gabriel Kahn, and Cyrus Shahabi. 2020. Crime-Forecaster: Crime Prediction by Exploiting the Geographical Neighborhoods' Spatiotemporal Dependencies. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 52–67.

[14] Xiaofeng Wang and Donald E Brown. 2012. The spatio-temporal modeling for criminal incidents. *Security Informatics* 1, 1 (2012), 1–17.

[15] Detian Zhang, Yuan Liu, An Liu, Xudong Mao, and Qing Li. 2017. Efficient path query processing through cloud-based mapping services. *IEEE Access* 5 (2017), 12963–12973.