Machine Learning: How ML Can Detect Crime in Charlottesville

CS4991 Capstone Report, 2023

Connor Goodall
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
cag3dmr@virginia.edu

ABSTRACT

In the wake of the shooting that happened in the UVA community a few months ago, many students may feel unsafe traversing the city of Charlottesville. To help students feel safer, my team created a machine learning algorithm to help forecast violent and non-violent crime rates in Charlottesville. We created this algorithm by splitting the crime dataset from the Charlottesville open data catalog into a training set and a test set. We utilized eight different regression models for the training set and tuned their hyper-parameters and chose the one with the lowest error rate as the model for the algorithm. The newly-created algorithm gave a distance of 0.016% between the predicted and actual violent crime rates in the test set and gave a distance of 0.85% between the predicted and actual non-violent crime rates in the test set. In the future, this project could be used in a visualization tool allowing students to see the safest path to take to their destination using our machine learning algorithm.

1. INTRODUCTION

On the night of November 13, 2022, a mass shooting occurred in the UVA community. Three people were killed and two others were injured. Four of the victims were members of the UVA football team. Thankfully, the perpetrator of this horrific crime was found and brought to justice.

Despite this, many students may still feel unsafe traversing the city of Charlottesville, walking to class, or living in their dorms. They might feel this way due to the fear that another shooting could occur in the future or that the UVA community is not well-protected against crime.

To help students feel safer in the community, my machine learning class team decided to make our semester-long project on helping the UVA community combat crime. We decided to create a machine learning algorithm that would help forecast violent and non-violent crime rates in Charlottesville. As students, we thought it would be beneficial if our algorithm could identify areas where crimes occur most often or could predict the likelihood of being a victim of a crime in a particular area. This would help students avoid these areas in Charlottesville in order to feel safer.

2. RELATED WORKS

Shah, et. al. (2019) suggested using machine learning and computer vision algorithms and techniques in order to assist police officers in preventing crime. Our project utilized the same machine learning algorithms and techniques as Shah, et. al. However, we used them to assist UVA students to avoid crime instead of assisting police officers to prevent it. In addition, we trained our algorithms for predictions that would be better suited for UVA students instead of police officers.

Pinto, et. al. (2020) suggested using classification with machine learning models

in order to help police officers understand the future scope of crimes. Our project utilized many of the machine learning models as Pinto, et. al. However, we used them for regression purposes rather than classification. In addition, the predictions our algorithms produced were better suited for UVA students than police officers.

Zhang, et. al. (2020) suggested using machine learning algorithms and techniques in order to predict crime hotspots for Chinese cities. Our project utilized many of the machine learning algorithms that Zhang, et. al. used. However, we used them to predict crime hotspots in Charlottesville rather than Chinese cities. In addition, the dataset and its variables used to train each machine learning algorithm was better suited for UVA students.

3. PROCESS DESIGN

To create the machine learning algorithm, my team needed to find a dataset that contained crimes in Charlottesville, feature engineer new variables in the dataset, clean the dataset, and select the model that was the best fit for the dataset.

3.1 Dataset

My team found a crime dataset of more than 24,000 entries from the Charlottesville open data catalog. It gave the record ID, offense, police incident ID, block number, the street where the offense occurred, agency that took the report, report date and time, and reporting officer.

3.2 Feature Engineering

My team created the violent crime and the non-violent crime variables. The violent crime variable included any crime that is murder, rape, robbery, or aggravated assault. The non-violent crime variable included all other crimes. We added the violent crime and non-violent crime variables to the original dataset. Then, we found the number of violent and non-violent crimes that happened in each area of Charlottesville and divided this number by the area's population to get the violent and non-violent crime rate variables.

My team used the Census Geocoder API and the Census Enterprise Area API to determine the area's population. We had to feed the Census Geocoder API the area's longitude and latitude in order to retrieve the area's census block. Then, we fed the Census Enterprise Area API the area's census block in order to retrieve the area's population.

To get the area's longitude and latitude, we used the street address from the original dataset and the Batch Geocoder for Journalists API. We transformed the street addresses into the format the Batch Geocoder for Journalists API wanted and fed the API the street addresses to get a dataset that contains all the coordinates and the full street addresses. If the full street address did not contain Charlottesville or any street in Charlottesville, we removed it from the dataset. We then added the longitude and latitude variables to the original dataset.

3.3 Data Cleaning

To clean the dataset, my team separated the dataset into three sets: the training set, the test set, and the validation set. The original dataset gave 80% of its data to the training set, 10% to the test set, and 10% to the validation set. Then, we removed the violent crime rate and the non-violent crime rate variable from both the training and test sets. We used a pipeline and column transformer to completely clean the training and test set for preparation of model selection.

3.4 Model Selection

My team tried eight different regression models on the training set for the violent and non-violent crime rates for each area in Charlottesville: linear regression, decision tree regression, random forest regression, stochastic gradient descent regression, support vector regression, lasso regression,

ridge regression, and gradient boosting regression. For each of these regression models, we hyper-tuned their parameters in order to create the best version of the model.

3.4.1 Violent Crime Rate

For the violent crime rate, the error rates for each of the regression models on the training set is shown below:

Table 1: Error Analysis for Violent Crimes

Linear Regression	Decision Tree Regression	Random Forest Regression	SGD Regression	SV Regression	Lasso Regression	Ridge Regression	Gradient Boosting Regression
0.073%	0.045%	0.034%	4.60%	0.14%	0.76%	0.074%	0.084%

Of these eight regression models, the random forest regression model was the best fit and was chosen as the final model since the average distance between the predicted violent crime rate and the actual violent crime rate was smaller compared to the average distance for the other regression models.

3.4.2 Non-violent Crime Rate

For the non-violent crime rate, the error rates for each of the regression models on the training set is shown below:

Table 2: Error Analysis for Non-violent Crimes

Linear Regression	Decision Tree Regression	Random Forest Regression	SGD Regression	SV Regression	Lasso Regression	Ridge Regression	Gradient Boosting Regression
2.20%	0.49%	0.61%	141%	2.24%	3.69%	2.22%	3.29%

Of these eight regression models, the decision tree regression model was the best fit and was chosen as the final model since the average distance between the predicted non-violent crime rate and the actual non-violent crime rate was smaller compared to the average distance for the other regression models.

4. RESULTS

For the violent crime rate for each area in Charlottesville, the error rate for the final model on the test set was 0.016%. This meant that the predicted violent crime rate for each area in Charlottesville and the actual violent crime rate for each area were 0.016% apart and that the final model was able to predict accurately on the violent crime rate for each area in Charlottesville.

For the non-violent crime rate for each area in Charlottesville, the error rate for the final model on the test set was 0.85%. This meant that the predicted non-violent crime rate for each area in Charlottesville and the actual non-violent crime rate for each area were 0.85% apart and that the final model was able to predict somewhat accurately on the non-violent crime rate for each area in Charlottesville.

5. CONCLUSION

Overall, our machine learning algorithm was able to semi-accurately predict the violent and non-violent crime rates for each area in Charlottesville. This will allow students who use our algorithm to determine what path that they plan to take is the safest. If our algorithm predicts a high violent or non-violent crime rate for a certain area, the student may decide to go a different way. This may prevent students from becoming a victim of a violent or non-violent crime and help them feel safer in Charlottesville by avoiding these areas with predicted high violent or non-violent crime rates. This project will end up making Charlottesville feel a little safer for its residents.

6. FUTURE WORK

In the future, our machine learning could be used in a visualization tool that allows students to see the safest path to take to their destination. The student would need to place into the tool where they are currently at and their destination. Our machine learning algorithm would then predict the violent and non-violent crime rates for every possible path by using the violent and non-violent crime rates for each area on that path. It would compare each path's violent and non-

violent crime rates with one another until it found the path with the lowest violent and non-violent crime rates. The visualization tool would then display this path so the student could see the safest path to take to their destination.

REFERENCES

Pinto, M., Wei, H., Konate, K. & Touray, I. (2020). Delving into factors influencing New York crime data with the tools of machine learning. *Journals of Computing Sciences in Colleges, 36*(2), 61-70.

Shah, N., Bhagat, N. & Shah, M. (2021). Crime forecasting: a machine learning and computer vision approach to crime prediction and prevention. *Visual Computing for Industry, Biomedicine, and Art, 4*(9), 1-14. https://doi.org/10.1186/s42492-021-00075-z

Zhang, X., Liu, L., Xiao, L. & Ji, J. (2020). Comparison of machine learning algorithms for predicting crime hotspots. *IEEE Access, 8*, 181302-181310. https://doi.org/10.1109/ACCESS.2020.3028420