Global Terrorism Preparedness Model

Springboard Capstone Project Report

May 9, 2024

Samuel Castillo

Table of Contents

Section

Supplement

Introduction

Over the last 70 years, the University of Maryland (in conjunction with the Pinkerton Global Intelligence Service) has collected detailed recordings of over 200k terrorist incidents and made this data accessible to the public through the Global Terrorism Database (GTD). Terrorism in this case is broadly defined as "the threatened or actual use of illegal force and violence by a non-state actor to attain a political, economic, religious, or social goal through fear, coercion, or intimidation.[1]"

The aim of this project is to employ predictive analytics based on the GTD to help governmental defense departments and independent intelligence agencies strengthen their overall prevention, preparation, and response to specific terrorist threats with greater success. The deliverable is an interactive dashboard that allows users to view the probabilities of success[2] for various threats given a description of the current social or political circumstances that may give rise to an attack. Essentially, the dashboard reveals the events of highest likelihood and vulnerability given the input to aid in strategic preparation. The predictive model faces some challenges due to class imbalance across various features, but the dashboard offers modular thresholds that assist in overcoming these obstacles. Overall, this project is a success, and it is just one of the many ways to harness the GTD as a fundamental source of predictive analytics in the field of global preparedness.

The Dataset

The initial dataset consisted of two csv files: records from 1970 to 2020 and records from 2020 to 2022. Combined, the total records amounted to just over 209k observations, each holding 135 features. Features included but were not limited to the following: datetime data; location details including geographic coordinates; various categories specifying target/attack types; numeric ranges of fatalities, injuries, etc.; Boolean features such as whether an event was successful for the perpetrators; details of the perpetrators where available; citation details of relevant news articles; and a description of the perpetrator's "motive" outlining the social and political circumstances that gave rise to the incident.

The dataset contained several null values for features that were unavailable or irrelevant for certain incidents. For instance, many features were added retroactively in 1998 and in real-time henceforth, causing some significant inconsistencies in the recorded detail of observations prior to 1998 relative to those after. In other cases, a field for 'second weapon type' may be blank simply because there wasn't a second weapon present.

---

[1] *Global Terrorism Database Codebook*, University of Maryland.

[2] *Success in this case is defined by 1) the infliction of death or injury or 2) the fulfillment of a perpetrator's motive (e.g., receiving a ransom payment).*

Overall, however, the other organizations involved in building the GTD took great care in ensuring its quality and consistency wherever possible, and users can view the publicly available GTD codebook as an indispensable resource in understanding the data properly.

Objective

The notebooks reveal multiple shifts in approach throughout the project. Though each new direction fell under the overall umbrella of the purpose stated in the introduction, the primary approach to capturing the most relevant feature relationships and definitions of target variables varied widely for the first several notebooks. The project began with an emphasis on certain major US cities and briefly incorporated multiple census reports and web-scraped population data to increase the feature volume and predictive power, but these approaches were either poorly suited for a machine learning task or too cumbersome to handle in the scope of this project.

The final objective begins to bud in Notebook 2.3 and comes to full fruition in Notebook 3.1. As mentioned in the introduction, it is defined as such:

Produce a model that predicts the probabilities of success for every possible terrorist incident given a user-defined description of the current political or social circumstances.

This question set the Boolean feature 'success' as the target and the string feature 'motive' as the primary predictor via an NLP analysis. Other features were of predictive importance as well, such as the target type and attack type categories.

Wrangling and Cleaning

The first step was to develop a complete understanding of each of the 135 features in detail via the GTD Codebook available in the "reference" folder: this involved noting the significance, ranges, data types, etc. for each feature. This was integral in the choice to trim certain features and keep others for analysis – several were redundant (e.g., numeric categorical representations of textual information) and others were not relevant to the question at hand (e.g., news sources, extent of property damage, etc.).

The next step was addressing duplicates, null values, and data types. As the GTD has been carefully assembled and reviewed, there were no purely duplicated entries – some entries were nearly identical, but they represented related incidents with evenly split metrics to capture holistic measures across multiple records.

There were, however, several null values to address. Most null values indicated a lack of detail for a given observation (such as missing perpetrator characteristics), but some nulls simply indicated an unrecorded metric and could sensibly be imputed as '0' (e.g., number of victims killed or injured). Other values were null placeholders or otherwise indicative of nulls, such as '-9' or 'Unknown'. Depending on the field in which these values were present, they were often filtered out altogether.

After filtering out records with significant and irreparable null values, the distribution of incidents over time was practically flat until the 1998 mark. Given the GTD's retroactive approach taken to pre-1998 recordings, this made sense. Since only a marginal proportion of observations after filtering took place before 1998, all pre-1998 observations were removed for the sake of consistency and consolidation.

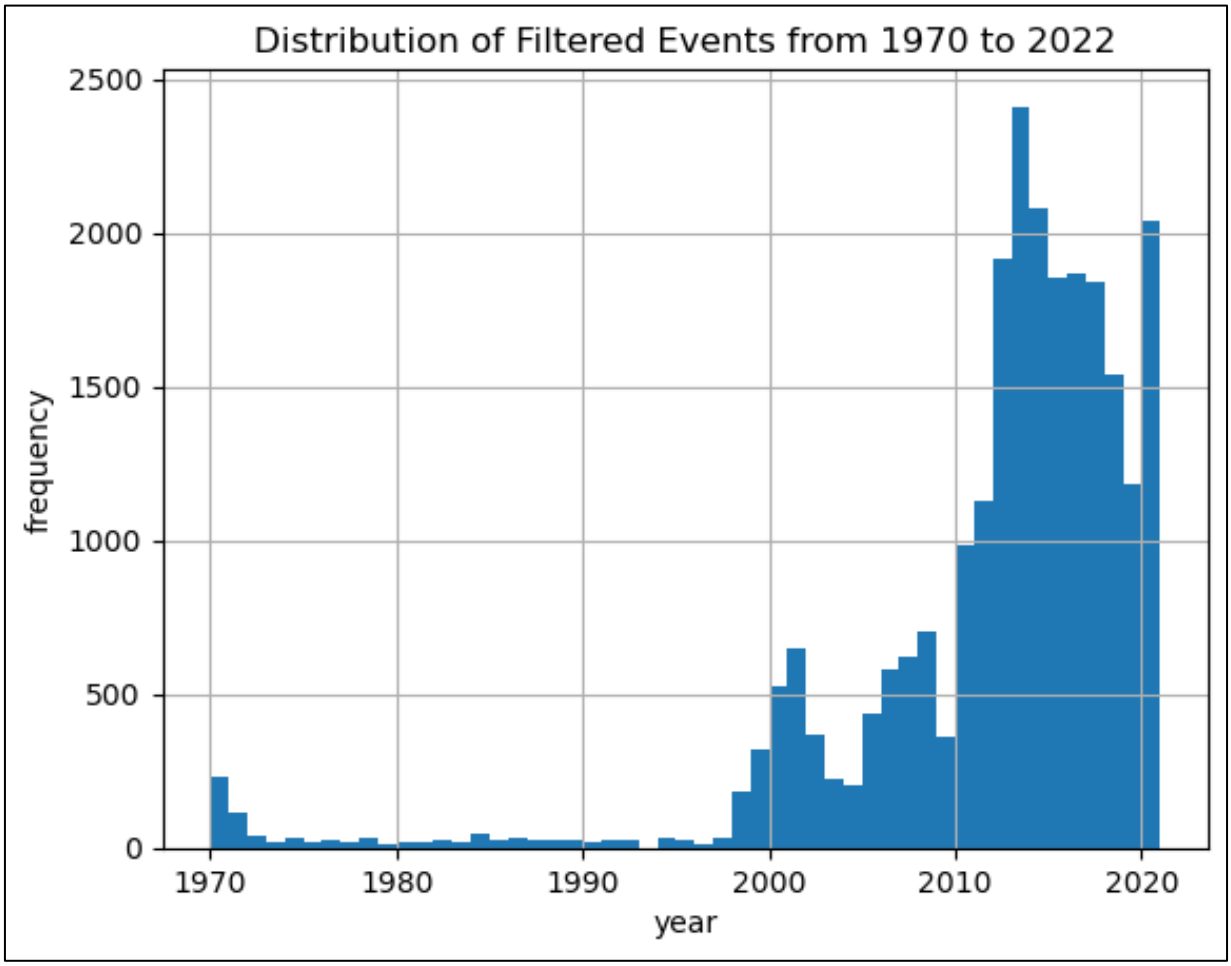

Figure 2. Distribution of Filtered Events from 1970 to 2022. *The filtered data demonstrates a sharp increase in recorded event frequency following 1998.*

The final dataset for preprocessing consisted of 10 features and 23764 observations from 1998 to 2022. The features and datatypes are recorded in the table below.

Feature	Data Type
Day	Integer
Month	Integer
Year	Integer
Motive	Object
Target Type	Category
Attack Type	Category
Weapon Type	Category
Number Killed	Integer
Number Injured	Integer
Success	Boolean

Table 3. Basic Schema of Final Dataset. *Most features are either integers or categories, but 'Motive' and 'Success' are variable object and Boolean, respectively.*

Exploratory Data Analysis

EDA required several iterations as the scope of the question took different forms. The first analysis involved distributions of timelines and features of events in major US cities in search of interesting patterns. This bred some opportunity for questions in further studies.

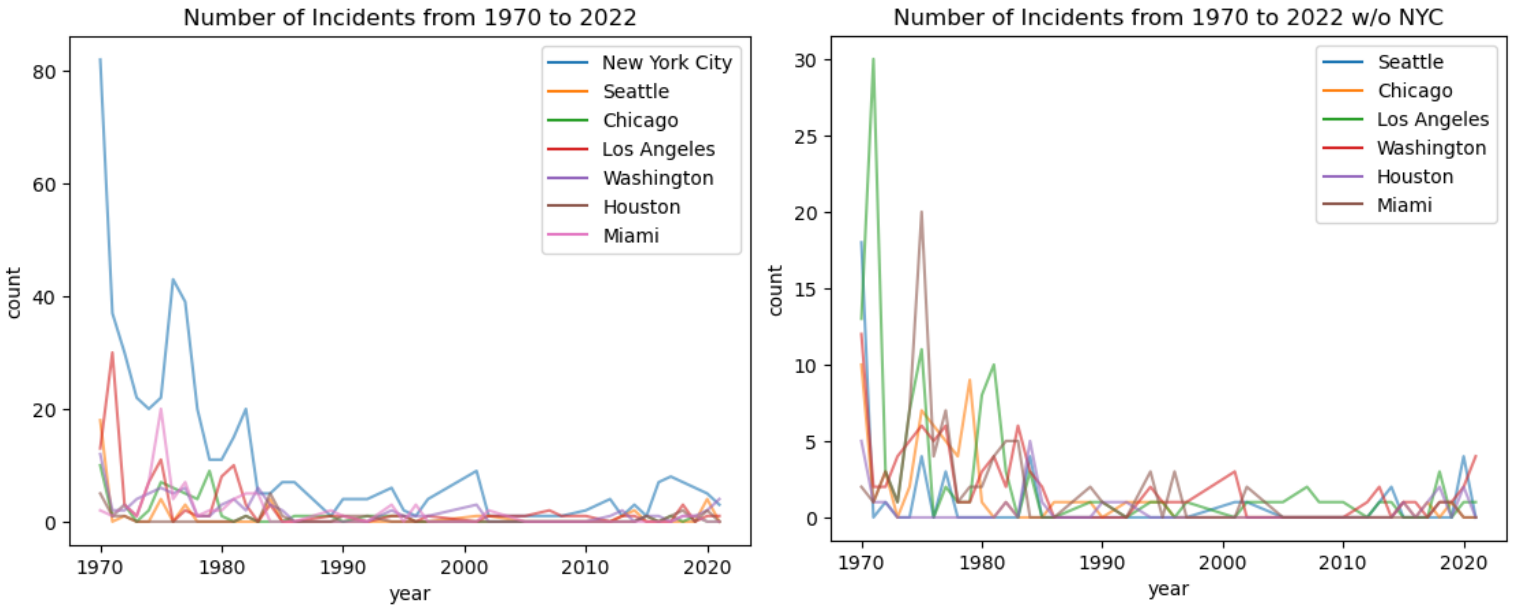

Figure 2. Timeline of Summed Incidents in the US over time with and without New York. *The relative distribution is much easier to detect in the latter image, as NY is the leader in summed incidents by a large margin.*

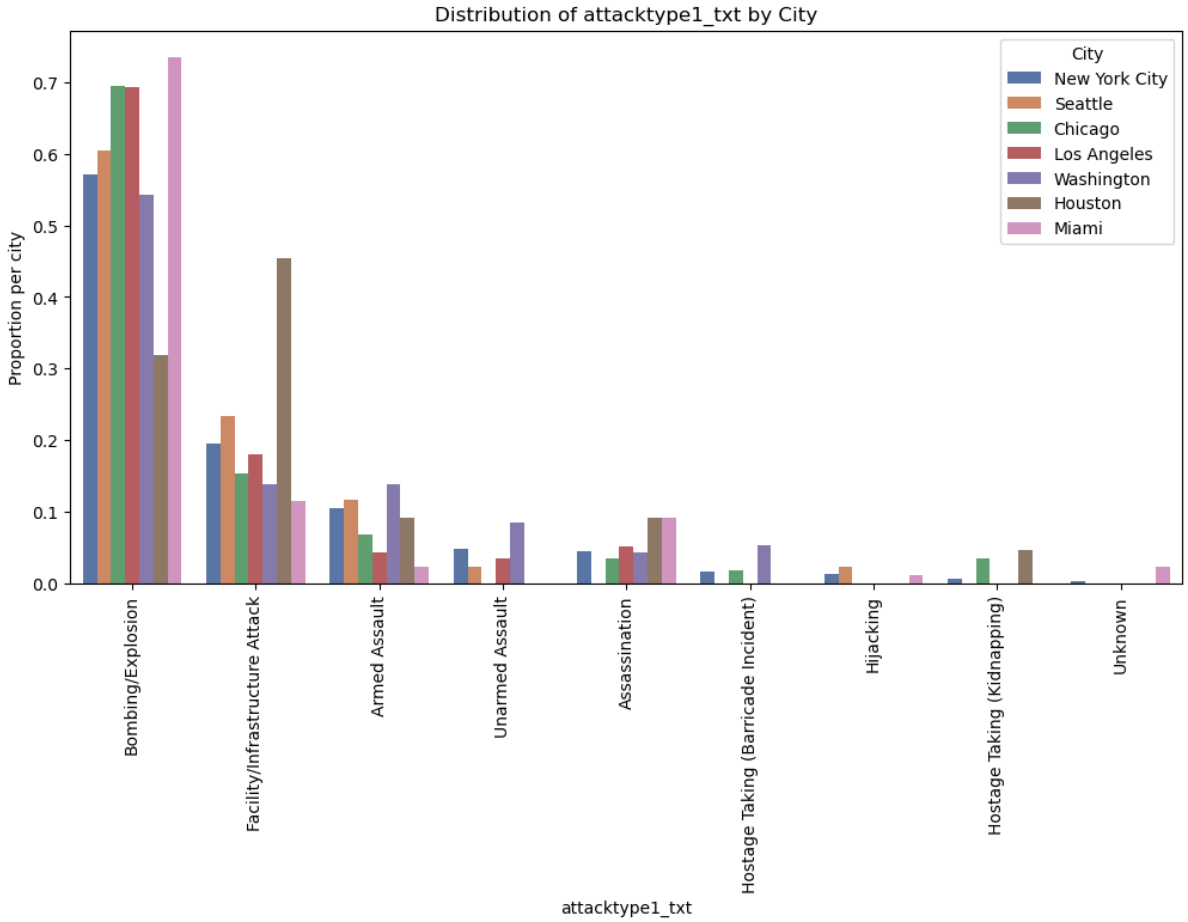

Figure 3. Proportion of Attack Types by US City. *A few observations of note include the distinct outlier of Miami for the proportion of bombing/explosion attacks and Houston for facility/infrastructure attacks. What factors may lead to these observations?*

In an attempt to form a more stable question and increase analytical capacity, external data was introduced to the original GTD data. This involved loading, cleaning, and concatenating hundreds of features from American Census Bureau surveys. The next analysis was an overview of several distributions and correlations between various features in the GTD and census features taken at the statewide level.

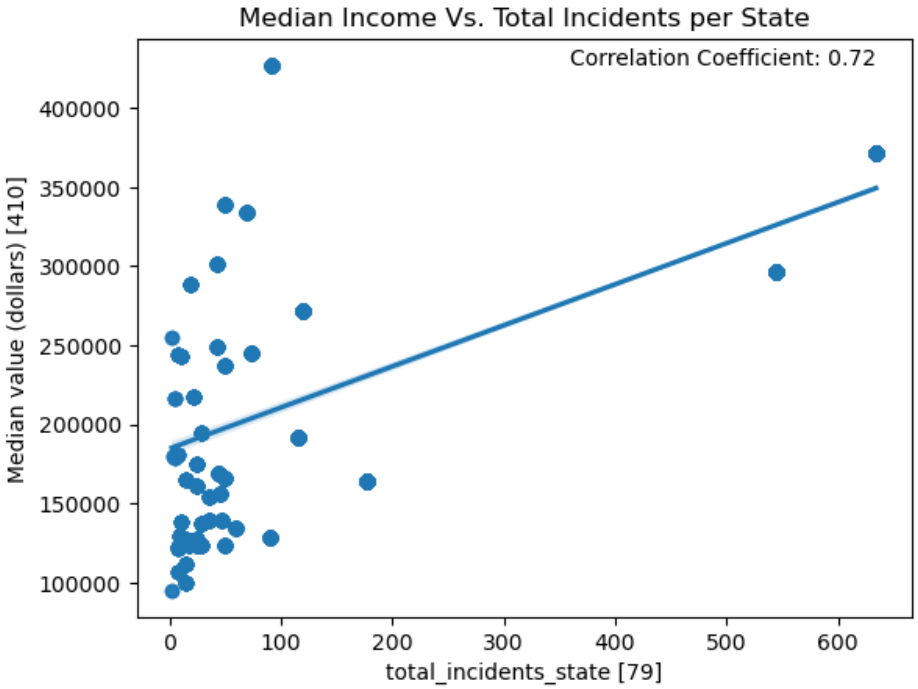

Figure 4. Median Individual Income Vs. Total Incidents per State. *According to American Census Survey data taken from 2010, there is a slightly positive correlation between the median individual income and the total incidents per state. This is not a likely foundation for further exploration as several factors could act as confounding variables in this case.*

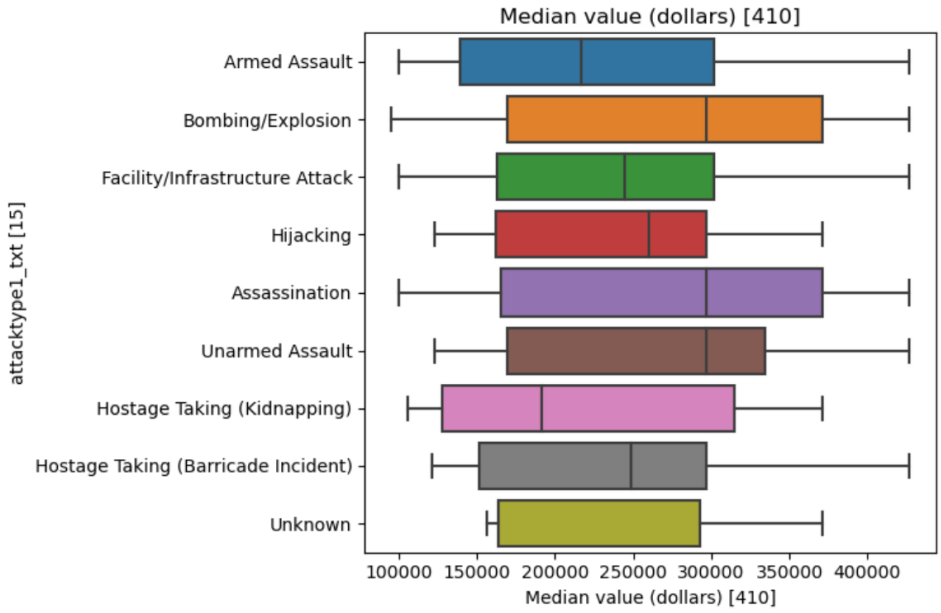

Figure 5. Distribution of Attack Types According to the Median Property Values per State. *According to census data, it appears that bombings, assassinations, and unarmed assaults are more frequent in states with a higher median property value. Hostage kidnappings are most common in states with lower property values.*

When this approach proved too inconsistent and unwieldy for a well-controlled ML task, The external data was left behind and the approach was consolidated to the features within the GTD itself. To maintain analytical capacity, this required an expansion to the global scale and exploration of the 'motive' feature, as it provided a wide variation of key details that would provide predictive power. Ultimately, this would require Natural Language

Processing (NLP) techniques. Analysis at this stage consisted of a few word clouds prior to NLP to get an idea of the distributions of word frequencies for various scenarios.

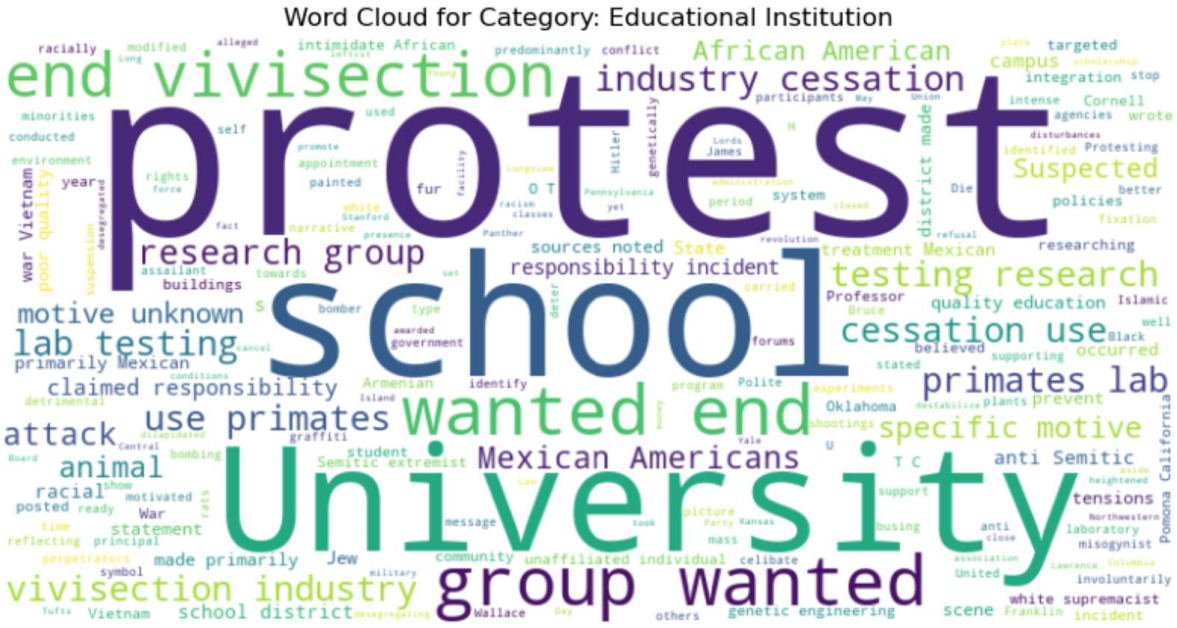

Figure 6. Word Cloud of 'Motive' Feature for Attacks on Educational Institutions in US. *The word cloud applies font size as a measure of frequency. Thus, it seems that most terrorist incidents on educational institutions are attempts at protesting against the use of animal research, vivisection, etc. This is a helpful visual to understand how word frequency can help identify patterns in string variables.*

Granted, using the 'motive' feature to predict target types and attack types in a traditional ML approach would have proven rather cumbersome, as the combinations of multi-level classes would have overcomplicated the classification model. Instead, the 'motive' feature could be used in conjunction with combinations of target type and attack type categories to predict probabilities for the single binary feature of 'success.' Given the wide range of possibilities to explore with this new approach, EDA led to the creation of a Tableau dashboard to simplify the process of viewing the ratios, timelines, and geographic locations of successful versus unsuccessful events of various circumstances.

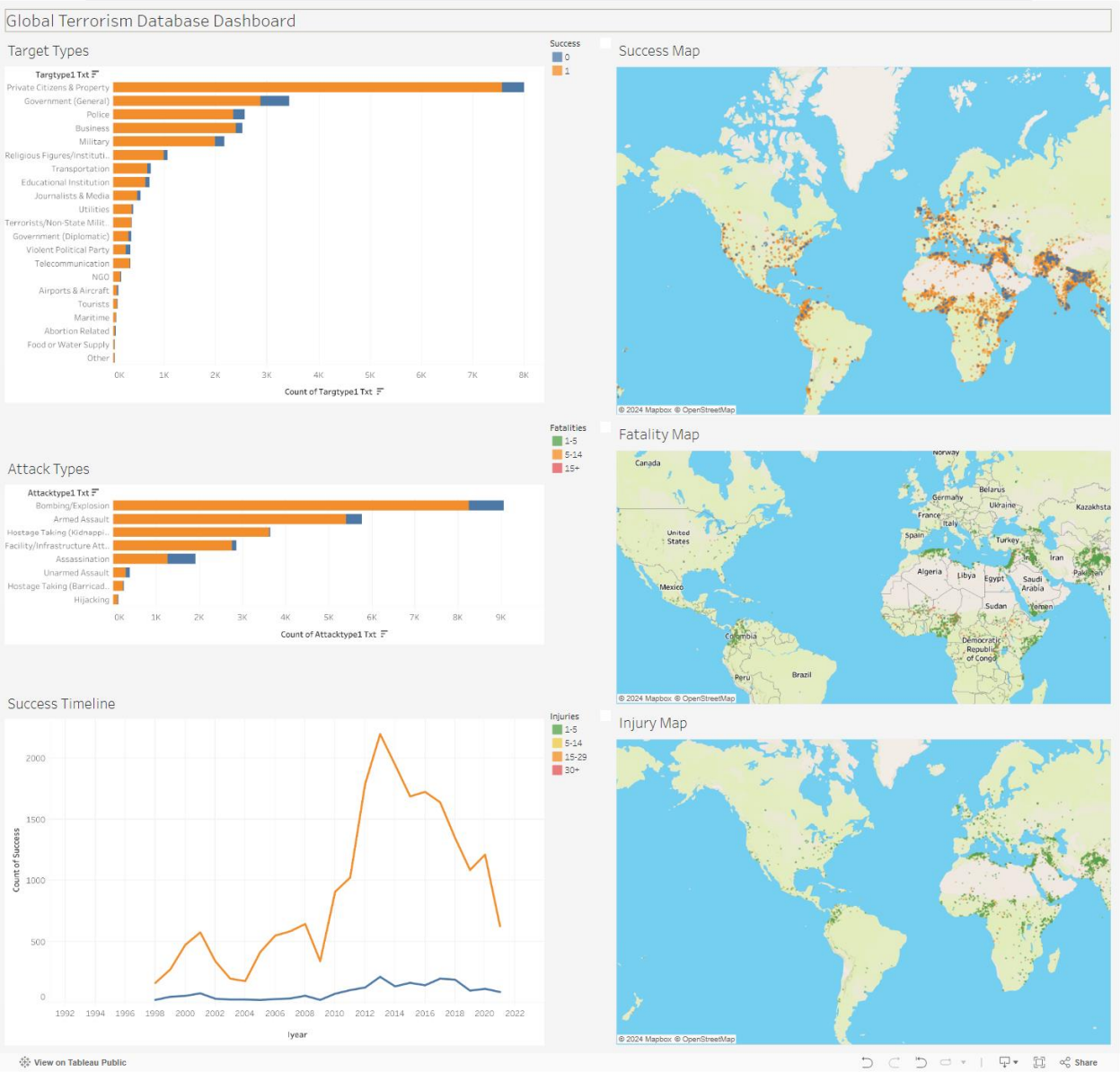

Figure 7. Tableau Dashboard of Incident Success with Maps. *This dashboard allows the exploration of success versus failure given specific target types, attack types, geographies, fatality/injury counts, or any combination thereof. Recorded incidents are successful due to survivor bias eliminating unsuccessful events, but different combinations of possibilities differ in the proportions of success, making this dashboard a useful explorative tool.*

Taken together, the success dashboard and categorical word clouds form the conceptual foundation for the predictive model moving forward.

Preprocessing

Once the objective was formally redefined as a natural language processing task using the 'motive' feature and a few categories, preprocessing was relatively straightforward. The first step was dealing with the numeric features: number killed and number injured. These features were heavily left-skewed as most recorded events (even if successful) result in zero or few deaths or injuries, as the goal is often property damage or simply public

recognition. To address this skew and make these features useful to the model, values were cut based on intuitive thresholds and placed into binned categories to even out the distribution (0, 1-5, 6-10, etc.).

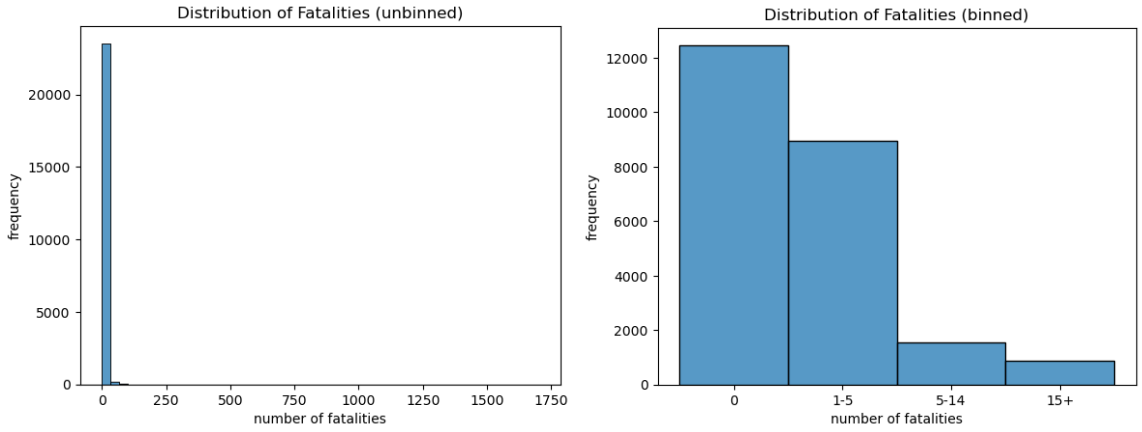

Figure 8. Distribution of Fatalities Before and After Ordinal Binning. *Though the distribution remains skewed after binning, it's much easier to visualize fatalities in discrete categories rather than in continuous form. The same applies to injury counts as well.*

The next features to process were categorical features: target type, attack type, and weapon type. Target type contained 21 levels ranging from 30 to 7991 instances. This imbalance was too broad to represent levels well in the model, so all classes below 200 instances were bagged into a broad "other" class to leave 15 levels total. Similarly, attack type contained 7 levels ranging from 121 to 9065 instances, and all classes below 1000 instances were bagged into the "other" class to leave 6 levels total. Weapon type values were not bagged as the feature was not guaranteed a place in the final model.

```
print('value counts before bagging attack types')
print(dfpr.attacktype1_txt.value_counts())
attack_mask = dfpr.attacktype1_txt.value_counts() > 1000
for idx, val in dfpr.attacktype1_txt.items():
    if attack_mask[val] == False:
        dfpr.loc[idx, 'attacktype1_txt'] = 'Other'
print('\n')
print('value counts after bagging target types')
print(dfpr.attacktype1_txt.value_counts())
```

```
value counts before bagging attack types
attacktype1_txt
Bombing/Explosion                  9065
Armed Assault                      5772
Hostage Taking                     3879
Facility/Infrastructure Attack     2861
Assassination                      1907
Unarmed Assault                     390
Hijacking                           121
Name: count, dtype: int64


value counts after bagging target types
attacktype1_txt
Bombing/Explosion                  9065
Armed Assault                      5772
Hostage Taking                     3879
Facility/Infrastructure Attack     2861
Assassination                      1907
Other                               511
Name: count, dtype: int64
```

Figure 9. Value Counts Before and After Bagging. *Bagging thresholds were set to eliminate the largest gaps between lower levels. All bagged values are considered together as 'Other' henceforth.*

After bagging, the *pandas* 'get_dummies' function was used to encode the three categories as they possessed no inherent order. However, the fatality and injury count categorical bins were ordered, so an ordinal encoder was applied for those features.

As an added element of analysis, the year, month, and day features were combined into a single 'date' feature, and the new 'date' feature was further engineered into a binary category indicating whether an attack took place on the weekend or during the week.

The last step of preprocessing involved NLP techniques on the 'motive' feature, as this was the crux of the whole project. Upon initial review of the 23764 values present, there were a few peculiar patterns: namely, several (if not most) of the entries began sociopolitical descriptions with the phrase "It was believed that…" to indicate uncertainty or subjectivity regarding the perpetrator's motive. Though that may be important for the GTD's integrity, repetitive meaningless phrases were entirely unhelpful to this project as NLP is based on word-frequency distributions.

After careful consideration, the word "believe/d" was removed entirely from the motive feature using regular expression functions to prevent washing out other results with repetitive use of a meaningful word. It was no issue to keep words like "it" or "was" as these are noted as stop-words and removed by default, i.e., words that add little to no semantic value. However, the word "believe" has stronger semantic value and would likely skew the results unnecessarily.

In the same pass, all digits were removed as well, as they can also add unnecessary washout to the frequency distribution.

```
pd.set_option('display.max_colwidth',None)
dfpr.motive
```

```
0
It was believed this attack was meant to intimidate UN inspectors and halt weapons inspections.
1                                                                                          It wa
s believed that the attack was a protest to election results that occurred on December 15, 1997.
2                                                                                          It wa
s believed that the attack was a protest to election results that occurred on December 15, 1997.
3
This attack was meant to terrorize civilians and the government ahead of continued peace talks.
4                                                    It was believed that this attack was one of a series o
f retribution killings that occurred following the death of the attackers' leader, Billy Wright.
```

Figure 10. Demonstration of 'Motive' Feature. *The phrase 'It was believed' (or similar) was extremely common in the 'Motive' feature. Since it adds no semantic value worth predicting, it was removed from each record.*

At this stage, a train-test-split was applied to prevent data leakage during the NLP stage. All steps henceforth were applied to training and testing data separately. Given the objective, the predictor (X) sets consisted of all features except the 'success' variable, and the target (y) sets consisted of the 'success' variable.

I used the *nltk* library to tokenize and lemmatize values based on a *WordNetLemmatizer* loaded from the library. Though it takes longer, lemmatizing was superior to stemming as it maintained a greater semblance of semantic significance. Then, a Tf-idf vectorizer was initiated with ngram range of (2,3), fit/transformed to the lemmatized values, and concatenated to the encoded categorical features from previous steps. The resulting output was too large to be stored in computer memory, so the project moved forward in sparse matrices.

The resulting dimensions of the training and testing predictor sets were (17823, 162926) and (5941, 162926), respectively. The target sets were naturally identical along the first dimension with only 1 feature in the second dimension: 'success.' The data was ready for modeling.

Model Evaluation

This project required the application of a binary classification algorithm to predict the probability of the 'success' variable returning positive given the predictors generated during categorical encoding and NLP. Vectorization generated hundreds of thousands of features for consideration, ruling out the possibility of several traditional models. Moreover, the 'success' class was subject to survivor bias, as most unsuccessful attacks were not recorded in the database because they had minimal social impact – thus, the target feature was heavily imbalanced.

Given the challenging circumstances of over 100k features and heavy class imbalance, three models were rigorously evaluated to find the most optimal path forward: Random Forest

Classifier, Linear Support Vector Machine Classifier, and Light Gradient Boosting Machine Classifier.

Evaluation consisted of two steps:

1) Randomized search cross-validation to determine the optimal hyperparameters for each model.
2) Comparison of confusion matrices, precision/recall/f1 score, and area under the precision-recall curve of each tuned model's performance on the test set.

A note on evaluation:

Given the class imbalance and objective at hand, a simple measure of accuracy (correct predictions/total predictions) was too elementary of an evaluation. The emphasis of the objective was to assist in terrorism preparedness, meaning the model should minimize false negatives – in regards to terrorism, agencies would much rather prepare for an event that does not occur than fail to prepare for an event that does occur. That said, no one can prepare for everything and should allocate resources strategically, so false positives were also significant. For this reason, f1 score and area under the precision recall-curve were the primary metrics of comparative evaluation.

Following hyperparameter optimization, the three models performed well on the test set. Random forest was slightly behind (and much slower) in performance compared to SVC and LightGBM, and the latter two were essentially tied for first.

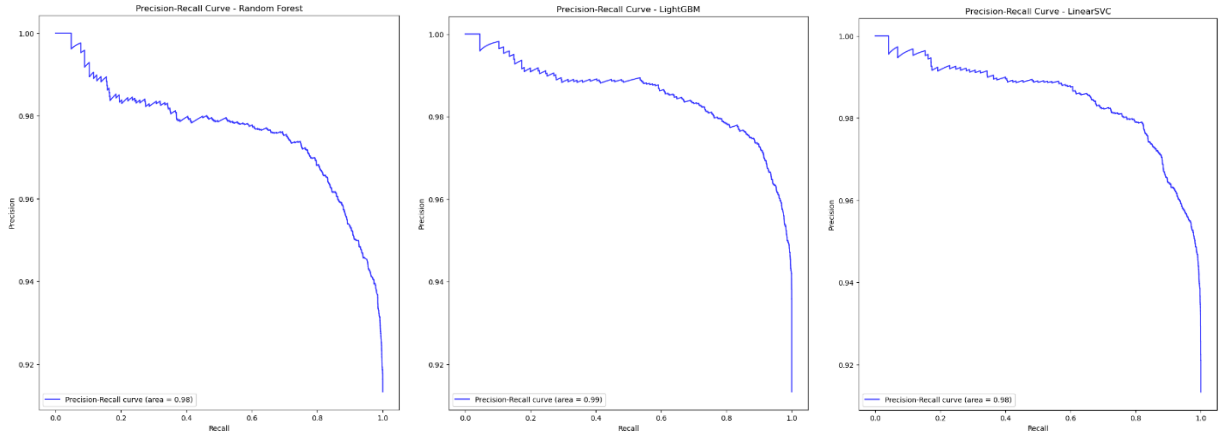

Figure 11. Precision Recall Curve Evaluation of Three Models. *Random Forest (L) demonstrates AUC of 0.98, Linear SVC (M) of 0.99, and LightGBM (R) of 0.98. Though not shown, LightGBM demonstrated a slightly higher ROC AUC and F1 score, hence the tie between LightGBM and LinearSVC.*

To decide between the two better models, more extensive cross-validation allowed further optimization of hyperparameters and re-evaluation. In the end, LightGBM won by a very slight margin.

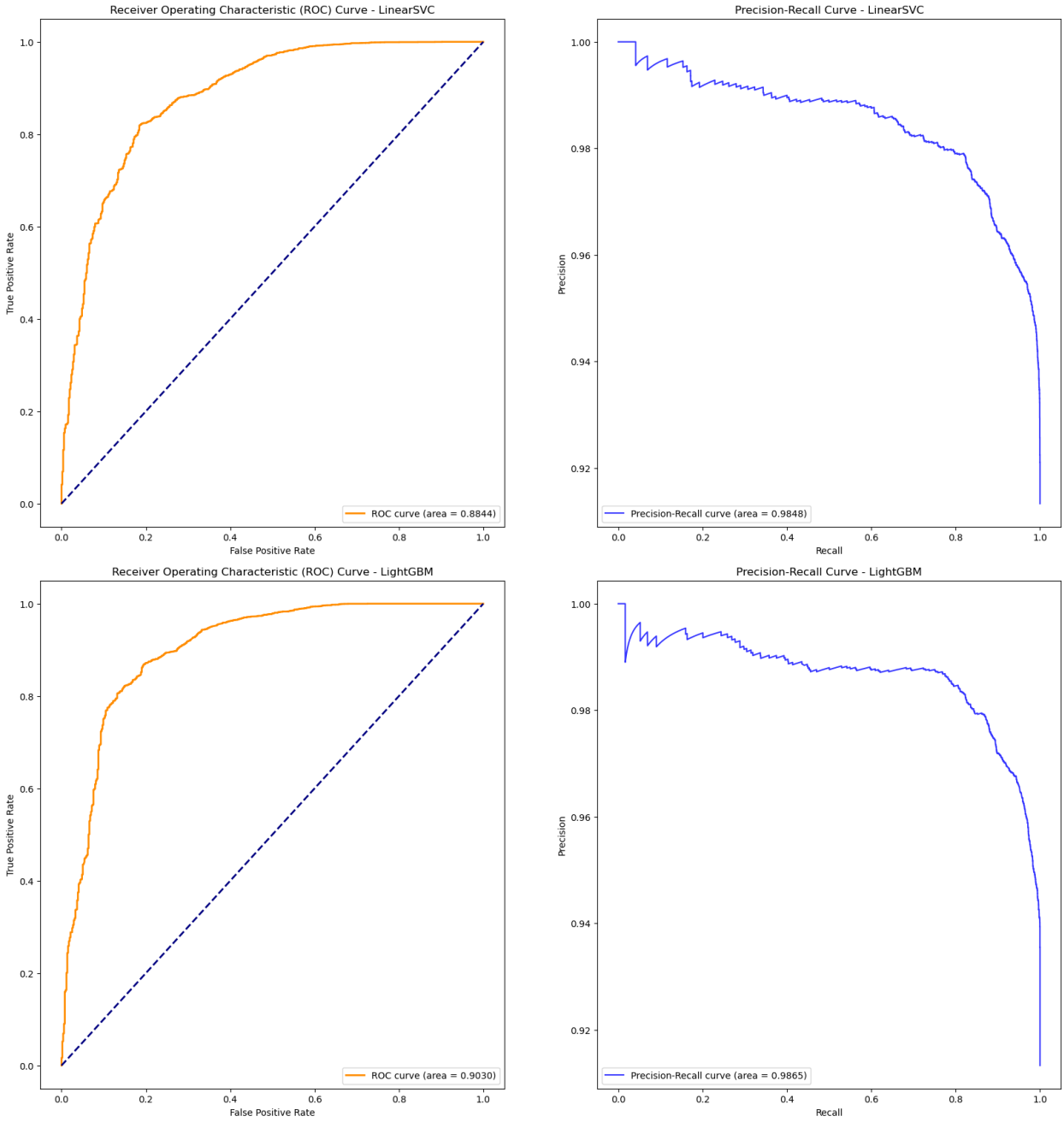

Figure 12. Final AUC Evaluation of Optimized Linear SVC versus LightGBM. *After extensive hyperparameter tuning, Linear SVC (above) demonstrated ROC AUC of 0.88 and PR AUC of 0.9848. LightGBM (below) demonstrated ROC AUC of 0.90 and PR AUC of 0.9865. Thus, the optimized LightGBM won by a slight margin.*

The figures above demonstrate excellent performance of the LightGBM model in binary classification of the 'success' feature based on the NLP word-frequency features and encoded categorical variables.

Product

The final LightGBM model was incorporated into an interactive format that allows users to view probabilities of success for various combinations of features based on their own input of the 'motive' feature. Essentially, any description of the social or political circumstances that could give rise to a terrorist incident constitutes a valid input.

The interaction looks like the following:

```
model()

Enter a potential suspect or affiliation and motive in your own words: Political Extremist student organization at Universit
y of Tennessee is protesting recent educational bill that hinders the development of government-funded charter schools
Tokenizing...
Lemmatizing...
Vectorizing...
Converting to sparse matrix...
Initializing LGBM...
Fitting LGBM...
[LightGBM] [Info] Number of positive: 21705, number of negative: 2059
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 1.014943 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 83407
[LightGBM] [Info] Number of data points in the train set: 23764, number of used features: 3134
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.913356 -> initscore=2.355322
[LightGBM] [Info] Start training from score 2.355322


Making Predictions...
Creating interactive dashboard...
Results ready for analysis!
```

Figure 13. User Interaction with Global Terrorism Preparedness Model. *The user enters an input (marked by red brackets) describing the current sociopolitical circumstances that may potentially give rise to a terrorist incident. The output is simply an echo of steps performed and ultimately an interactive dashboard.*

The results are then projected onto a simple dashboard built using the Plotly library. The dashboard allows modular thresholds for frequencies and probabilities to help users bypass washout due to imbalanced values. Here is an example output using the motive given above:

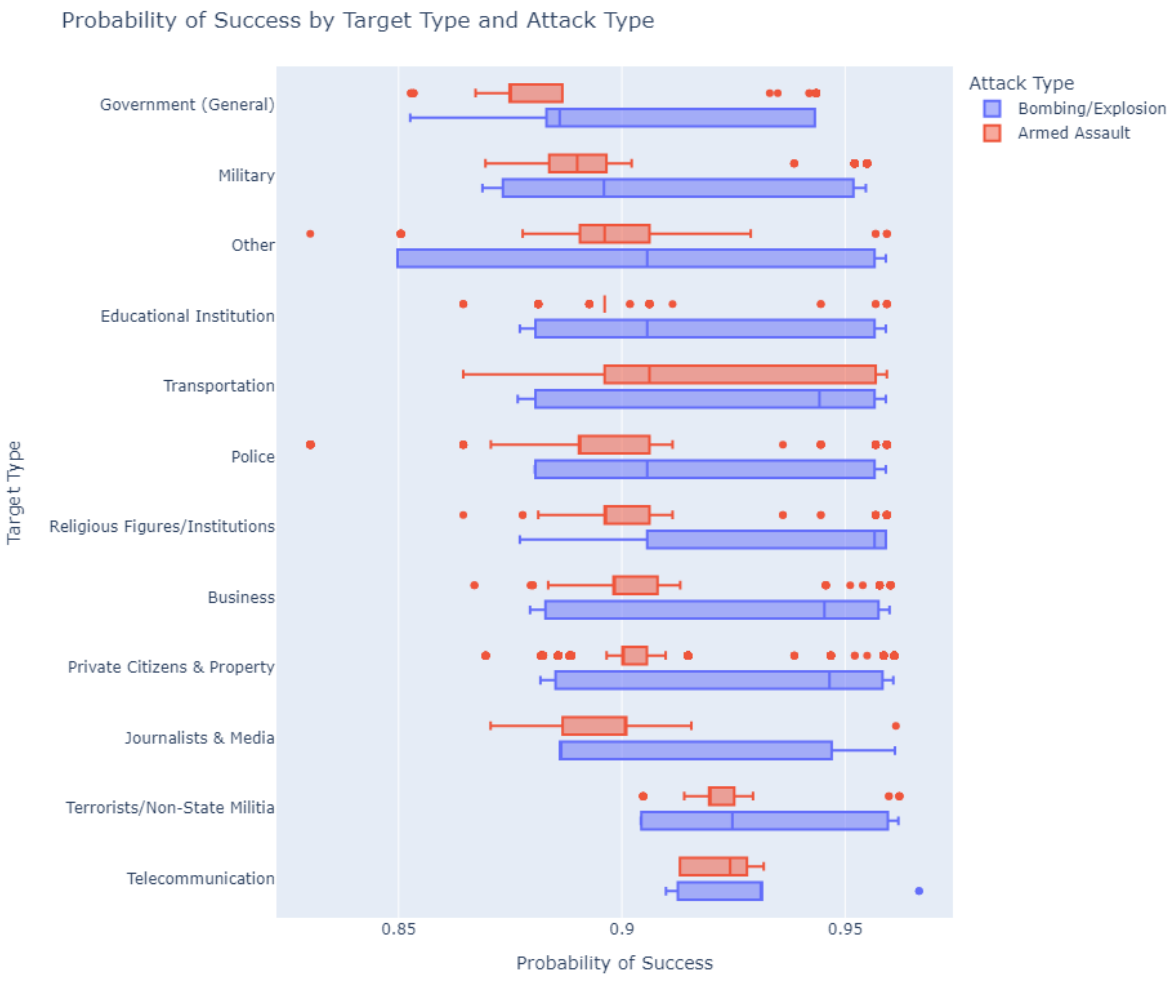

Figure 14. Global Terrorism Preparedness Model Dashboard Output. *The dashboard displays probabilities of success for incidents involving various target types and attack types given the motive input from Figure 13. The user can set thresholds to account for some values simply being more common (such as bombings) to more easily visualize possibilities of interest.*

The tool has plenty of room for improvement, but it is fully functional. Future iterations will involve a more complex dashboard that includes more features, a more sophisticated approach to class imbalance within each feature, and more innovatively engineered features based on other data (such as geography or time-series).

Conclusion

In summary, this project provides an interactive implementation of the Light Gradient Boosting Machine Classifier to predict probabilities of terrorist events through NLP analysis of a user-defined description of the current sociopolitical climate. Future iterations may involve more extensive hyperparameter tuning via Bayesian Optimization or even new models built on convoluted neural networks. Overall, this project is a solid foundation for future predictive models built on the Global Terrorism Database to aid in threat prevention and response.