Transportation Research Record
2020, Vol. XX(X) 1–9

DOI: 10.1177/ToBeAssigned
journals.sagepub.com/home/trr
SAGE

Spatio-Temporal crash Prediction: Effects of Negative Sampling on Understanding Network-Level crash Occurrence

Peter Way[1], Jeremiah Roland[1], Mina Sartipi[1], and Osama Osman[2]

Abstract

In projects centered around rare event case data, the challenge of data comprehension is greatly increased due to insufficient data for deriving insight and analysis. This is particularly the case with traffic crash occurrence, where positive events (crashes) are rare with, in most cases, no data set existing for negative events (non crashes). One method to increase available data is negative sampling, which is the process of creating a negative event based on the absence of a positive event. In this work, four negative sampling techniques are presented with varying ratios of negative to positive data. These types of techniques are based on spatial, temporal, and a mixture of the two types of data, with the data ratios acting as class balancing tools. The best performing model found was with a negative sampling technique that shifted temporal information and had an even 50/50 data split, with an F-1 score of 93.68. These results are promising for ITS applications to inform of potential crash locations in an entire area for proactive measures to be put in place.

Introduction

The definition of negative sampling is dependent on the use case. It is most popular among Natural Language Processing (NLP) and numerical research environments (*1*), where negative sampling is a method of data selection and filtration. However, negative sampling is also being sought after in the research of traffic patterns and crashes, as well as other smart city applications (*2*), (*3*). The benefit of using negative sampling on a traffic crash dataset is to gain a more thorough understanding of the different factors that contribute to vehicle crashes. It is critical to understand the various available types of negative sampling techniques, and which of these types may be best applied to answer a given research question. The positive samples explored in this study are traffic crash records from Hamilton County, Tennessee, and include temporal and spatial specifics from the crash location, as well as weather and roadway specifications. Various negative sampling techniques are explored in this paper, most of which are temporally and spatially reliant.

Using records of past crashes and their corresponding data, our goal is to predict where crashes are most likely to happen. Vehicular crash data does not have 'negative' occurrences recorded. Therefore, they are a perfect choice for exploration of negative creation. The focus of this paper is the use of negative sampling as a balancing tool to outline the different methods available for creating non-crash records, as well as discussing which methods yield the best results in crash prediction.

The remainder of this paper is structured as follows: Section provides the literature survey on negative sampling techniques, Section covers the study data, the machine learning algorithm used, and the negative sample creation processes, Section displays the results on predicting crashes using different negative sampling methods, Section discusses our results in details, and Section concludes our findings and future works.

Related Work

Roadway Safety Projects

A Convolutional Long-Short Term Memory (ConvLSTM) (*4*) was applied to a study by (*5*) with vehicular accidents in Iowa, between 2006 and 2013. Data included crash reports from Iowa Department of Transportation, rainfall data, roadway weather data, and specific roadway geometric data including speed limits, AADT, and traffic camera counts. Due to the area of study being so expansive, a five kilometer square per block grid layout was constructed to cover the state for crash prediction. Training data included 2006 to 2012 reports, with 2013 being reserved for testing. Tests involved

[1]Department of Computer Science and Engineering, University of Tennessee at Chattanooga, Chattanooga TN, USA
[2]Department of Civil and Chemical Engineering, University of Tennessee at Chattanooga, Chattanooga TN, USA

Corresponding author:
Mina Sartipi, mina-sartipi@utc.edu

predicting locations for the next week based on data provided by the previous week. ConvLSTM results out performed all baselines in prediction accuracy. Additionally, the team's predictive model correctly predicted crashes resulting from the heavy snowstorm seen on December 8th in 2013.

An initiative led by the United States Department of Transportation (USDOT) means to partner crowd-sourced data provided by Waze and safety policy decisions to help predict vehicular accidents (*6*). A combination use of Classification and Regression Trees (CART) and Random Forest models are used for crash prediction. The study included 6 months of accident data from Maryland, which was paired with the corresponding Waze alerts, if any existed. While not perfectly identical, specific temporal and spatial event patterns created by the model were fairly similar to the actual crash records. Additionally, the model tended to under predict crashes in the early morning hours, while over predicting crashes for high-commute periods. This prediction trend was attributed to the historical spread of accidents. Of particular note is the model's ability to predict Waze alerts for minor crashes, which are crashes that are significant enough to inhibit standard traffic flow, but not serious enough to report to law enforcement.

The effects of weather conditions on daily crash occurrences were analyzed using a discrete time-series model by (*7*). An integer autoregressive model was used for modeling count data with time interdependencies. The model was then built using daily car crash data, weather data, and traffic exposure data from three Netherland cities: Dordrecht, Utrecht, and Haarlemmermeer. Loop detector data was used to collect daily vehicle counts for each road segment of the major road networks. From this, each city region's major roadway network had its day-to-day total kilometers driven calculated. Weather data was also collected for the three cities and was broken up into specific weather instances. For example, precipitation was broken up into duration, daily amount, rain intensity, etc. This type of deaggregation was done for wind, temperature, sunshine, precipitation, air pressure, and visibility. It was discovered that several weather variables were significant in relation to accident occurrence.

State-specific Safety Performance Functions (SPFs) for rural interstates and rural 2-lane roads was used by (*8*). With these functions, 20 segments of each type of road were identified with the highest Potential for Crash Reduction (PCR). A Cost Benefit Analysis (CBA) was then performed using Crash Modification Factors for the types of crashes occurring. This resulted in an index that normalized the safety benefit of all roadway classes based on the cost of implementation. Model Minimum Uniform Crash Criteria was also used along with Knee Airbag Deployment models for identifying and classifying accident data. Once the road segments that had the highest PCR values were identified, CBA was used to identify which sites would provide a return on investment and in ranking the segments deserving treatment.

An investigation into roadway accident likelihood and severity in Athens, Greece was conducted by (*9*). The study consisted of roadway accident data from a main roadway in Athens from 2006 to 2010. A logistic regression model was developed for analyzing traffic patterns and performing predictions. It was found that the severity of roadway accidents was heavily influenced by the logarithm of traffic density, the vehicle type, and the accident type. Additionally, it was found that traffic density was the only statistically significant variable when the traffic accident data was split into peak and off-peak hour accidents. Furthermore, it was found that traffic volume was the only variable with a statistically significant impact on traffic accident likelihood.

Negative Sampling on Traffic Crash Predictions

A case study was conducted in (*2*) on predicting traffic crashes by comparing the results of four different classification models. In this study, a method of generating non-crash data was performed and called negative sampling. For each positive example (crash), the value of only one feature (hour, day, or road ID) of the crash record was changed. The resulting sample was then checked for a positive (match found) or negative (no match found) result. Once all negative sampling process was completed, the team concluded the study with triple the number of negatives than positives, roughly a 75/25 split of data. The study concluded that an ideal performance can be achieved with a neural network model and 3:1 ratio of negative to positive data. The results of the study suggest that a data split that maintains the nature of crashes as being rare events is required to achieve acceptable prediction accuracy.

The team of (*3*) performed similar tests with crash prediction and negative sampling. Antoine et al. created their negatives through a process akin to brute force. Time and location information of the crashes in their dataset were examined and every possible combination was generated, keeping only 0.1% of these newly created negatives. This method resulted in 2.3 million negatives for their dataset. While not explicitly stated, this ratio was likely chosen to reach a desired data imbalance factor, as the authors say "this corresponds to a total of 2.3 million examples with a data imbalance reduced to a factor of 17." This study is an exemplary frame of reference dataset balancing. Even with a method of producing an extraordinary amount of negative samples, it is impractical to utilize all negatives as they could lead to an imbalanced data problem.

The interactions between roadway geometry, weather, and traffic data on the occurrence of vehicular crashes was studied by (*10*). The area of study was on the mountainous freeway of highway I-70 in Colorado, consisting of 301 crash records and 880 non-crash records. These non-crash records were obtained from an automated vehicle identification system,

where each non-crash entry was extracted for situations where no crash occurred 2 hours before the reported crash entry. This type of negative sampling is best described as a simple temporal shift, which is outlined later in this work.

Negative Sampling on Language Based Projects

Negative sampling has been more commonly used in projects not related to crash analysis, such as Natural Language processor projects seen in (*1*). Four strategies of negative sampling (local sampling, distance sampling, uniform sampling, and refined sampling) were studied for language processing applications. These four strategies were applied in exploration of Yahoo! question and answer community forums. *Local Sampling* negatives are those close to the existing positive sample by some given measure of approximation. This measure is able to be linguistically handled by comparing how similar individual words are, or based on how similar different grouped words were to other groups of words. *Distance Sampling* negatives are those as distinct and different from the positive entries as capability allows. This ensures the data is correctly clustered in the given space of study. *Uniform Sampling*, simply said, is the random selection of negatives within the given space. This ensures that the entire space to be explored is represented equally, without preference to similarities or lack there of. *Refined Sampling* was defined as the combination of Local and Distance styled sampling, with the pursuit of a model capable of spanning clustered embeddings within a single category, as well as different categories. The study in (*1*) also outlined some rules for negative samples; negative samples should be i) as similar as possible to positive samples to increase the model's discriminative abilities, ii) as different as possible to positive examples to avoid feeding the model conflicting information, and iii) representative of the entire space of negative samples. In other words, negative samples need to be distinguishable enough from the positive samples in a dataset to be considered their own entity, yet still represent the same fundamental information as positive samples to be used in conjunction with them.

Three unique divisions of negative sample creation were also presented by (*11*). They are presented as incompatible relations, domain specific rules, and random samples. *Incompatible Relations* are relations that always, or almost always, conflict with the relation wished to be extracted (*11*). In the case of traffic crash prediction project, an incompatible relation would be between generation of negative samples that exactly match current positive samples. If a generated negative sample has a certain time, date, and location, then positive samples cannot exist with the same time, date, and location, as there cannot be a non-crash where an crash was recorded. *Domain Specific Rules* are negative samples that are highly specific towards the particular data one is exploring (*11*). Similar to the above mentioned example, one cannot have a non-crash with the same time and location parameters. *Random Samples* deal with marking some current data as negative evidence. An example would be from the work of (*1*), where the authors dealt with Yahoo! QnA posts. In this work, random negative sampling is described as the process of randomly sampling questions from a pool of all answered questions across all categories of questions.

Negative sampling has the unique ability to generate inherently present but not inherently available data, that data being non-crash records. While there exists a plethora of studies related to vehicle crash prediction, very few implement negative sampling as a method of data enhancement. Our contributions to this issue are outlining the different methodologies behind negative sample generation, identifying which methods yield worthwhile results, and which methods perform best in real world implementation.

Methodology

Data

Figure 1 shows a flow chart of the process for data collection, data processing and aggregation, and modeling and visualization. The original crash data comes from the Hamilton County Emergency Services Department, dating from October 2016 to present day, and is updated daily. After data cleaning, the dataset consisted of roughly 61,000 crash entries. This dataset consists of the latitude and longitude of the crash, the time of the crash, and the crash severity (e.g., no injury, injury, mass casualty). The temporal and spatial information of the crashes are used to retrieve the weather and roadway geometries of the crash. Roadway geometrics were provided by E-Trims, a database from Tennessee Department of Transportation containing information on Tennessee roadway networks. Roadway data, such as those seen in Table 1, were collected by providing E-Trims with the Latitude and Longitude coordinates of the traffic crashes. Weather was collected via DarkSky, a Python API that collects data from many different weather sources and returns the most suited weather source related to the location provided. All weather data was collected by providing DarkSky the Latitude, Longitude, date, and time information of each traffic crash record.

Table 1. Data Features Used in Study

Variable	Description
Crash	Binary variable for crash occurrence
Hour	Hour of entry
UnixTime	Unix timestamp representation of entry
DayFrame	Time frame of day entry occurred (see Table 3)
WeekDay / WeekEnd	Binary variables representing weekend/weekday
Clear/Cloudy/Rain/Fog/Snow	Binary weather
RainBefore	If there was rain 1 hour before the crash
GridBlock	Spatial aggregation of the study area
Grid Col / Grid Row	Column and row of grid within grid layout (see Figure 2)
Highway	If there was a highway going through the GridBlock
Land Use Mode	Type of surrounding area (Ex. Commercial, Urban, etc)
Road Count	Count of roadways within GridBlock of entry

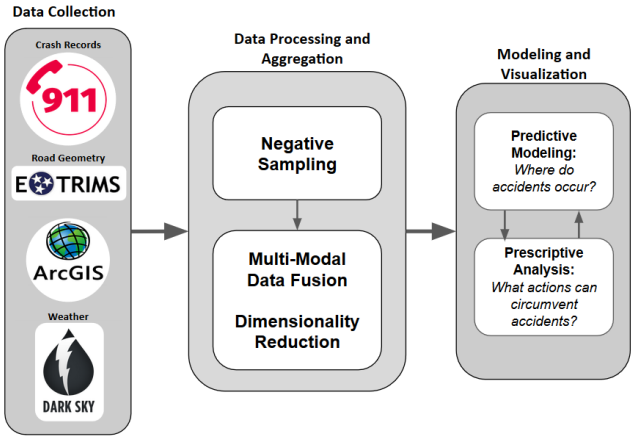

Figure 1. Illustration of Data Collection and Processing

All variables used in the negative sampling procedures and the creation of the given data set are shown in Table 1, along with a brief explanation of each variable. Grid Blocks, one of the variables used throughout this paper, refers to the image spaces seen in Figure 2. Each block seen is a Grid Block covering a 0.2 square mile area. The orientation of the grid block layout is matched to the orientation of Hamilton County's roadway network. Note that within the image, bolded black lines represent major interstates/highways. Additionally, white segments in the image convey bodies of water whose Grid Blocks are ignored in model creation/testing.

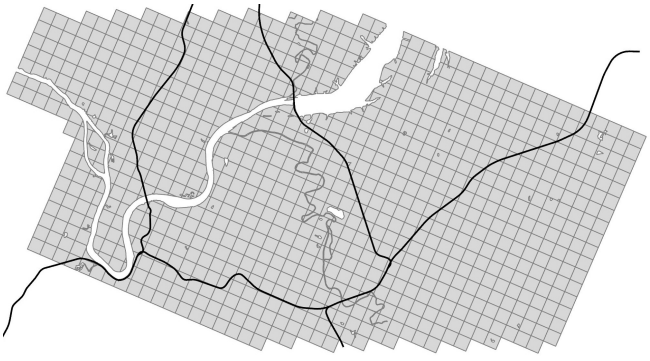

Figure 2. Grid Layout of Hamilton County used in Testing.

Originally, the project had no spatial aggregation and attempted to predict crashes for each roadway segment along the roadways in Chattanooga. This lead to massive over-prediction issues and excess noise, leading to the necessity for spatial aggregation. Initially, the grid blocks covered a .25 square mile area following the footsteps of (*12*). Further testing with grid blocks covering a .2 square mile area yielded superior model performance.

Machine Learning Model

A simple Multilayer Perceptron (MLP) acts as this work's machine learning model. Research into the various models used for crash prediction has shown that different regression style models examine traffic flow differently, and as such, lead to varying results (*13*). An example of this previous research shows that Poisson distribution proved valuable in crash frequency analysis relating to crash frequency modeling. Poisson also prevailed over traditional linear regression in highway safety applications (*14*). Additionally, Negative Binomial models are useful in exploration of crash severity, as shown in previous works (*15*). Furthermore, Negative Binomial models can be used for crash counts in datasets where over-dispersion occurs (*16*). In such cases, Negative Binomial models are more fitting than Poisson models. Ordered logit/probit models are commonly applied, although usage of these highly depends on the levels of injury severity (*15*). Within previous binary level injury severity studies, many research teams chose to apply binary logistic modeling (*17*), (*18*), (*19*).

When attempting to determine which analysis method would best fit our data and project (*20*), we conducted several different types of testing. Some techniques were based on conventional regression, while others were based on machine learning. Their results were overall lackluster. Before utilization of an MLP, select K Best testing (*21*) was applied for possible dimension reduction. When compared to the standard results of the MLP, the results of the various Select K Best tests underperformed in both accuracy and area under the curve, with K ranging from 5 to 25. These statistics were demonstrated by an accuracy range of 66.53-77.35% and an AUC range of 50-83.03%. Additional tests consisted of Naive Bayes, resulting in 62% accuracy, and a standard accuracy score test provided by Sklearn, resulting in 67.76%. Due to the lackluster testing results mentioned above, a standard MLP Model (*22*) was chosen for our study's machine learning technique. We use labelled inputs for classification prediction, which MLPs are suitable for. Furthermore, MLPs are very flexible with the use of data, which is extremely beneficial to our study as our dataset is very complex and intricate. MLP networks are comprised of an input layer, an output layer, and at least one hidden layer between the two. The details of the architecture used by our model are displayed in Table 2. Initially, compilation was provided by binary cross-entropy, which is particularly useful for binary results and classification. However, it was found that MSE (mean squared error) provided a significantly lower loss score with only a 2% cost in accuracy. Sigmoid acted as the model's activation function, as sigmoid is particularly useful for probability predictions because it limits a prediction model's output to a range of 0 to 1.

Table 2 displays the basic structural layout of the MLP model. Note in the *Node* column, a specific numerical value is provided for the number of nodes used for the sake of

Table 2. MLP Neural Network Architecture

Layer	Location	Type	Node	Activation
1	Input	Dense	# of Variables	Sigmoid
2	Hidden	Dense	# of Variables - 5	Sigmoid
3	Hidden	Dropout (0.1)	-	-
4	Hidden	Dense	# of Variables - 10	Sigmoid
5	Output	Dense	1	Sigmoid

simplicity. For the different tests performed for this study, it was decided to have a method in place where instead of manually adjusting how many variables would be used for the three layers, a simple subtraction equation was put in place to set the number of nodes per layer based on the number of variables supplied to the model. Note that this method of automated node count per layer requires there to be no less than 10 variables present for the model to use. Lastly, the training and testing split was 70% and 30%, respectively.

Creating Negative Samples

When a dataset consists solely of positive examples, attempts at discovering important features are impeded. Initial prediction trials for our work (*20*) utilized solely positive examples, leading to a high count of false positives, representing that the model was predicting crashes occurring at some location when in fact none did. Thus, even when many entries exist for crashes, attempts in prediction may fail. The results of (*2*) introduced the idea of implementing a negative sampling procedure for generating non-crash records. The procedure involves changing a single value of an crash record (hour, date, location) and checking if there is a matching crash record for the newly altered record. For example, if an crash occurred in hour 4, a new random hour was chosen between 0 and 23, excluding hour 4 for that day (*2*). The newly altered record was compared to all other crash records in the dataset to find any possible match. If no match was found, then the newly altered record was saved as a negative sample (non-crash). This process was repeated for every single crash entry in the dataset, and was done for each of the other two variable entries (date and location). This resulted in an increase in their dataset containing roughly 3 times more negative samples than positive samples. This process of negative sample generation was somewhat followed in previous implementation of our project (*20*), as at the time of initial implementation no roadway specific information, such as Road ID, was available.

During the creation of negative samples, it is possible that the acquired ”non-crash” record is actually a crash that is missing from our dataset. However, this possibility is negligible as starting in late 2016, Hamilton County Emergency Services Department began to actively take records of all 911 calls received in a day and format them in a consistent manner.

After completing preliminary trials with the negative sampling implemented, issues arose with accurate crash forecasting. Specifically, there was an under-prediction of crashes leading to high false negative counts. Therefore, it was decided to take a different approach to negative sample generation. In the negative sampling types described below, the process of negative sample generation was repeated 9 times for each crash record in an attempt to reach a 90/10 split in data (90 percent non-crashes, and 10 percent crashes). The concept for a greater number of non-crashes came from an article written by (*23*) that discussed the importance of having a greater amount of negative examples of an event class scenario when the positive examples of the specific event are rare by nature. Given the rarity of crashes occurring, the concept of maintaining the rare nature of a crash’s occurrence holds true for this project, thus the particular 90/10 method.

Sampling Types To examine the different effects that different negative sampling types had on model performance, four distinct types of negative sampling methods were created. These methods involved shifting spatial points, temporal points, or a mix of the two. **Temporal Shift** involves shifting either one or both of the temporal variables (Hour and Weekday), while freezing the spatial variable (Grid Block). **Grid Fix** is similar to Temporal Shift, in that the spatial variable is not changed when creating negative samples. However, each record in the crash list was examined and the hour and date of the record was changed. **Spatial Shift** involves shifting the Grid Block of the crash entry, while freezing the temporal variables. **Total Shift**, or Random Negatives, involves changing the hour, date, and grid block of the entry. This type of negative sampling is most similar to the negative sampling technique used in (*3*).

Negative Ratios In (*24*), an examination of traffic crashes is performed in Utah, exploring the importance of enough negative samples to clearly convey the rare occurrences of crashes, but not so many as to create a severe class imbalance. This imbalance leads to heavy bias toward the higher count occurrence, skewing prediction results. Conversely, training a model with an even split of non-crash and crash data may instruct the model that crashes and non-crashes occur with the same level of frequency. Now that the idea of varying ratios of negative to positive data has been introduced, the varying splits utilized by the aforementioned data may be explored further.

Original Modeling Split (66% - 33%): The negatives created at this stage of research were greatly inspired by (*2*), but included shifting the hour or date variable as at the time of initial creation, the 911 Project did not have roadway information. **Increased Negative Sampling Split** (75% - 25%): This split was inspired by the team of (*2*), as location negative samples were added to the dataset. For these, the location, date, and time variables for an crash record were changed. **Even Split** (50% - 50%): The even split was built upon the increased negative sampling split. The negative samples were scanned and every 3rd negative sample was

retained, effectively cutting each negative sample case in thirds while retaining the original span of the negatives created. **'Rare' Circumstance Split** (>90% - 10 %): This type of split was used to see how much of an impact an overwhelming amount of negative samples would have on model performance and crash prediction, while retaining the 'rarity' of crash occurrence. The 90/10 split only applies to the Temporal Shift and Spatial Shift negative sampling types, as the methods used to generate those negatives created an excessive amount of negatives.

Table 3. DayFrame Time Coverage

DayFrame	Hours Covered
DayFrame 1	0 - 4 and 19 - 23 (Overnight)
DayFrame 2	5 - 9 (Morning rush)
DayFrame 3	10 - 13 (Lunch hours)
DayFrame 4	14 - 18 (Evening rush)

Further Data Splitting Table 3 shows the different hourly aggregations that each DayFrame covers. This temporal aggregation came from the process of removing highly specific variables from the dataset to simplify the prediction process. Attempting prediction with highly specific temporal and spatial variables led to over-prediction. The different hour splits were selected based on crash trends seen in Figure 3, with DayFrame 1 (orange) covering the overnight hours, DayFrame 2 (green) covering the first spike in the Figure starting at hour 5, DayFrame 3 (blue) covering the second spike starting at hour 10, and DayFrame 4 (purple) covering the final spike starting at hour 14.

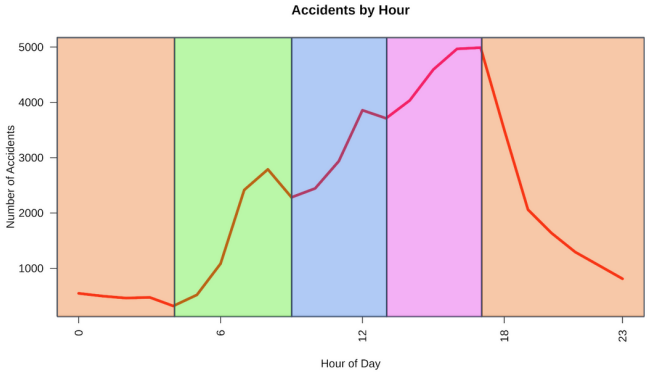

Figure 3. crashes by Hour and Weekday. Distribution of DayFrame hours encompasses the various crash trends throughout the day.

When conducting tests using the different negative sampling techniques, the terms "cut" and "full" are used in regard to negative samples. Full refers to the entire set of negative samples created through the respective method, while cut refers to a trimmed version of the negatives. This trimmed version was obtained based on the aggregated temporal variable DayFrame, see Table 3. For example, if a method of negative sampling produced 2 negatives, each negative's Hour variable was aggregated into the DayFrame variable, which values represent certain hour intervals of the day. Once properly aggregated, if the two created negatives have the same DayFrame, Date, and Grid Block, then one of the negatives are dropped so only 1 negative entry with that specific DayFrame, Date, and Grid Block remains. This was done to better represent the raw data as well as simplify the model's input variables.

All variables used for each of the data types and splits in the study were scored with a feature importance test utilizing the *ExtraTreesClassifier* algorithm. The most commonly occurring important variables are listed as follows: precipIntensity, Unix, Hour, Grid Block, Grid Col, Road Count, DayFrame, Land Use Mode, WeekEnd, Cloudy, RainBefore, and Clear. Note that some of these variables repeated in different importance slots across all the data types and splits. These twelve recurring variables represent the core identifying trends the model interprets when analyzing crash occurrence and expects when predicting crashes.

Results

In this section, we have used the following metrics to analyze the performance of the given models using True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN), where TP refers to correctly predicted crashes, FP refers to incorrectly predicted crashes, and so on. In regards to crash prediction, TP represents when the model predicts an crash correctly, TN represents when the model predicts a non-crash correctly, FP represents when the model predicts an crash when there was not one, and FN represents when the model predicts a non-crash when there was an crash.

$$Recall = \frac{TP}{(TP+FN)}$$
$$Precision = \frac{TP}{(TP+FP)}$$
$$F1\ Score = 2 * \frac{(Recall*Precision)}{(Recall+Precision)}$$

For the model testing results presented here, the performance of the model is based upon the recall, precision, and F1 scores. Recall refers to the percentage of correctly predicted crashes amongst all actual crashes. Precision is the ratio of correctly predicted crashes to all of the predicted crashes. F1 score is the weighted average of recall and precision, and the higher the value the better.

The results in Table 4 display our first attempts at negative sampling, consisting of the Original Modeling Split ratio of 66-33, and the Rare Circumstance Split of 90-10. These test results act as a baseline for comparison against the ratio tests seen in Table 5.

Table 4. Best Performing Test Runs for Negative Sample Datasets

NS Type	Train Acc	Train Loss	Test Acc	Test Loss	AUC	Recall	Specificity	Precision	F1 Score
Cut GridFix (66/33)	95.13	0.043	94.83	0.043	0.967	90.5	96.5	91	90.75
Full GridFix (66/33)	94.62	0.046	94.62	0.046	0.84	43.4	98.8	75	54.99
Cut Random (66/33)	94.55	0.046	94.39	0.046	0.957	78.7	98.3	92	84.83
Full Random (66/33)	67.97	0.040	68.02	0.214	0.84	81.6	66.8	18	29.49
Temporal Shift (90/10)	96.19	0.032	96.12	0.032	0.92	50.4	99.2	81	62.14
Spatial Shift (90/10)	99.65	0.003	99.66	0.003	0.789	37.5	100	100	54.55

Discussion

Ratio Tests

As mentioned in the previous section, recall, precision, and F1 scores are the ultimate determining scores for how well the crash prediction model performed. By using these metrics, the model results can be more confidently reported and a better understanding behind what a model is actually doing be gained.

When exploring classification data problems, there are many options on what exactly the ratio between positive and negative records should be. Considering the performance differences between the data splits, only 75%/25% and 50%/50% splits are explored here. Figure 4 displays a direct comparison between all the 50/50 and 75/25 model performances for correctly predicted crashes (top half) and non-crashes (lower half). Note the trend for 75/25 split models correctly identifying negative occurrences, but falling behind the 50/50 split models in positive event classification. Meanwhile more 50/50 split models correctly identified positive entries than 75/25, with a lower percentage of (yet not entirely atrocious) accurate classification when dealing with negative entries.

All of the above mentioned testing presented the even split producing the highest F-1 score, precision, and recall. As with the previous testing, the cut version of the grid fix negatives performed better overall in predicting crashes, resulting in **Cut GridFix (50/50)** yielding the highest values for F-1, precision, and recall. This is quite contrary to the original hypothesis regarding the necessity for a higher count of negative entries to represent the rarity of crashes in daily occurrences. Therefore we emphasize that the specific negative sampling technique and ratio of data must be deciphered for each unique research situation.

Limitations

The most severe limitation of this study is a lack of data, mainly resulting from the way crashes are reported. The majority of the specific data for crashes, such as the number of individuals involved, the drivers' ages, psychological profiles, intoxication levels, the age and conditions of vehicles, etc. are not available for analysis in our area of study, that being Chattanooga, Tennessee. Furthermore, with only three years of crash data available for analysis, there is the possibility of an insufficient amount of data to learn from.

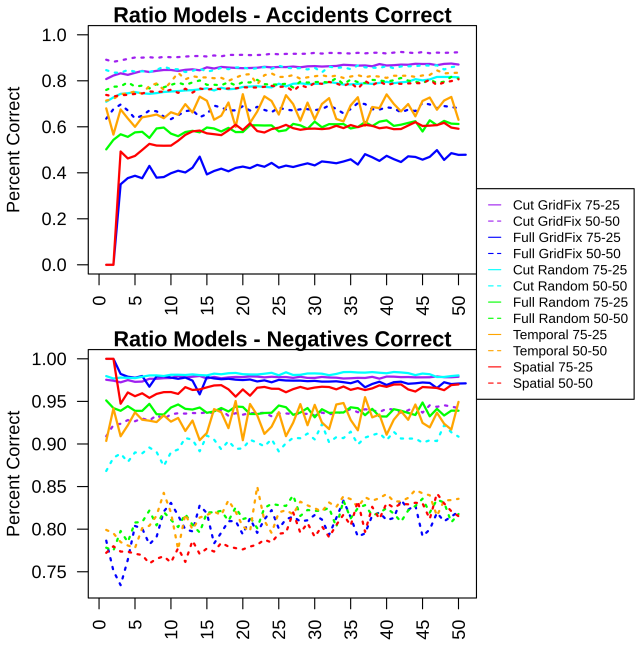

Figure 4. Percentages of Positive and Negative Entries correctly predicted via Ratio Models over 50 training cycles. Note that the x-axis represents the 50 training cycles.

Due to the previously mentioned chaotically random nature of crashes, a significantly additional amount of data is needed for a complete analysis of crash occurrence and cause.

Conclusion

Historically, negative sampling had been explored primarily in natural language processing projects or numerical research environments. However, the utilization of negative sampling is now being sought after in the research of traffic patterns, crashes, and various smart city research projects. This paper explored several negative sampling techniques, many of which take into account both temporal and spatial concerns that previous research into negative sampling had not addressed. It was found that fixing the Grid Block parameter and altering the Hour and Date variables produced the best result in predicting traffic crash records, with an F-1 score of 93.68, a precision score of 95, and a recall of 92.4. Thus, it can be stated for this application and data, that a temporal

Table 5. Ratio Test Runs for Best Performing Negative Sample Datasets.

NS Type	Train Acc	Train Loss	Test Acc	Test Loss	AUC	Recall	Specificity	Precision	F1 Score
Cut GridFix (75/25)	94.81	0.046	94.84	0.044	96.6	87.1	97.9	94	90.42
Cut GridFix (50/50)	93.75	0.056	93.24	0.057	96.6	92.4	94.2	95	93.68
Full GridFix (75/25)	85.13	0.115	84.96	0.114	86.3	47.9	97.1	84	61.01
Full GridFix (50/50)	75.08	0.171	75.19	0.167	83.1	68.1	82.1	79	73.15
Cut Randoms (75/25)	94.89	0.044	94.74	0.044	95.7	81.5	98.0	91	85.99
Cut Randoms (50/50)	89.05	0.083	88.62	0.083	95.2	86.4	90.9	90	88.16
Full Randoms (75/25)	85.48	0.107	85.44	0.106	89.6	61.2	93.9	78	68.59
Full Randoms (50/50)	81.58	0.135	81.37	0.132	89.3	80.9	81.8	81	80.95
Temporal Shift (75/25)	86.86	0.096	87.05	0.093	91.4	63.1	94.9	80	70.55
Temporal Shift (50/50)	83.12	0.128	83.53	0.122	90.5	83.5	83.6	84	83.75
Spatial Shift (75/25)	87.68	0.091	87.56	0.093	90.1	59.1	97.0	87	70.39
Spatial Shift (50/50)	82.11	0.130	81.09	0.130	89.3	80.6	81.5	81	80.80

shift with an even split between negatives and positives is the optimal route for a better crash prediction performance.

Of additional note is the robustness of the negative sampling methods outlined in this paper, as no one method is strictly locked to only work in a given environment. Additionally, the accident prediction project itself is robust as it uses easily gathered and widely available information that the majority of cities and counties would have access to. Should additional traffic crash projects attempt any negative sampling methods described in this paper, they would simply need the crash records to have hour, date, and location information. Currently the manner in which crashes are reported for our area of study, Chattanooga Tennessee, does not include more specific information regarding crashes, such as driver specific information, vehicle specific information, and traffic volume for Chattanooga itself. While this is a general limitation for the project, it also proves a valuable benefit to the project, as the lack of highly specific roadway, vehicle, or driver information means that, as stated previously, the majority of cities and counties would be able to implement this model in their area of study with relative ease.

As future implementations of negative sampling are performed, more detailed information would be added to the dataset as a whole. In particular, additional information regarding the crashes themselves would be greatly beneficial to their analysis. Additionally, using variable importance scoring would likely provide additional insight into the more important and meaningful variables available for analysis. Finally, further spatial aggregation measures could be implemented to enhance the project, such as those discussed by (*25*).

End of Paper Special Sections

Authors' Note

Peter Way, Department of Engineering and Computer Science, University of Tennessee at Chattanooga.

This research was supported in part by grants from NSF-US Ignite-1647161, as well as additional support from the city of Chattanooga.

Correspondence concerning this article should be addressed to Mina Sartipi, Department of Engineering and Computer Science, University of Tennessee at Chattanooga, TN 37403.

Contact: mina-sartipi@utc.edu

Acknowledgements

We would like to thank the City of Chattanooga, Hamilton County Emergency Communications District, Tennessee Department of Transportation, and Chattanooga Department of Transportation for supplying data for this research and valuable discussions. Furthermore, we would like to acknowledge the contributions to this project by Dr. Eric LaFlamme from Plymouth University for his help in different analysis techniques for our data and negative sampling. We also extend gratitude to the NSF-US Ignite-1647161 for partially supporting this project.

Author Contributions

The authors confirm contribution to the paper as follows: study conception, design and data collection and manipulation: P. Way, J. Roland; analysis and interpretation of results: P. Way, J. Roland, M. Sartipi, O. Osman; draft manuscript preparation and alteration: J. Roland, M. Sartipi, O. Osama. All authors reviewed the results and approved the final version of this manuscript.

Declaration of conflicting interests

There are no conflicts of interest to list for this manuscript.

Funding

This project was partially funded by the NSF-US Ignite-1647161.

Data Accessibility Statement

Data used in this project was provided to us by the Hamilton County Emergency Communications District. A subset of this data can be accessed on the site https://www.chattadata.org/Public-Safety/Police-Incident-Data/jvkg-79ss.

References

1. Sama, M., M. Saeidi, T. Togia, and R. Kulkarni. The Effect of Negative Sampling Strategy on the Performance of the Deep Structured Semantic Model. *Proceedings of the 1st International Conference on Natural Language Processing and Information Retrieval*, 2017.
2. Yuan, Z., X. Zhou, T. Yang, J. Tamerius, and R. Mantilla. Predicting traffic accidents through heterogeneous urban data: A case study. In *Proceedings of the 6th International Workshop on Urban Computing (UrbComp 2017), Halifax, NS, Canada*, Vol. 14. 2017.
3. Hébert, A., T. Guédon, T. Glatard, and B. Jaumard. High-Resolution Road Vehicle Collision Prediction for the City of Montreal. In *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019, pp. 1804–1813.
4. Sainath, T. N., O. Vinyals, A. Senior, and H. Sak. Convolutional, long short-term memory, fully connected deep neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4580–4584.
5. Yuan, Z., X. Zhou, and T. Yang. Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018, pp. 984–992.
6. Flynn, D. F., M. M. Gilmore, E. A. Sudderth, et al. *Estimating Traffic Crash Counts Using Crowdsourced Data: Pilot Analysis of 2017 Waze Data and Police Accident Reports in Maryland*. Tech. rep., John A. Volpe National Transportation Systems Center (US), 2018.
7. Brijs, T., D. Karlis, and G. Wets. Studying the effect of weather conditions on daily crash counts using a discrete time-series model. *Accident Analysis and Prevention*, Vol. 40, No. 3, 2008, pp. 1180–1190.
8. Davis, E. C., E. R. Green, N. Stamatiadis, G. A. Winchester, R. R. Souleyrette, and J. Pigman. Highway Safety Manual Methodologies and Benefit-Cost Analysis in Program-Level Segment Selection and Prioritization.
9. Yannis, G., A. Theofilatos, and A. Ziakopoulos. Investigation of road accident severity and likelihood in urban areas with real-time traffic data. *Traffic Engineering and Control*.
10. Ahmed, M. M., M. Abdel-Aty, and R. Yu. Assessment of interaction of crash occurrence, mountainous freeway geometry, real-time weather, and traffic data. *Transportation research record*, Vol. 2280, No. 1, 2012, pp. 51–59.
11. Generating Negative Samples - DeepDive, 2017. http://deepdive.stanford.edu/generating_negative_examples.
12. Mohler, G. and M. D. Porter. Rotational grid, PAI-maximizing crime forecasts. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, Vol. 11, No. 5, 2018, pp. 227–236.
13. Theofilatos, A. and G. Yannis. A review of the effect of traffic and weather characteristics on road safety. *Accident Analysis and Prevention*, Vol. 72, 2014, pp. 244–256.
14. Abdel-Aty, M. A. and A. Radwan. Modeling traffic accident occurrence and involvement. *Accident Analysis and Prevention*.
15. Khattak, A., L. Jun, and Z. Meng. Highway Safety Manual: Enhancing the Work Zone Analysis Procedure. *Transportation Research Record: Journal of the Transportation Research Board*, 2015.
16. Lord, D. and F. Mannering. The Statistical Analysis of Crash-Frequency Data: A Review and Assessment of Methodological Alternatives. *Transportation Research Part A: Policy and Practice*, Vol. 44, 2010, pp. 291–305. doi:10.1016/j.tra.2010.02.001.
17. Weng, J. and Q. Meng. Analysis of Driver Casualty Risk for Different Work Zone Types. *Accident Analysis and Prevention*.
18. See, C. F. Thesis: Crash Analysis of Work Zone Lane Closures with Left-Hand Merge and Downstream Lane Shift. *University of Kansas*, 2008.
19. Li, Y. and Y. Bai. Development of crash-severity-index models for the measurement of work zone risk levels. *Accident Analysis and Prevention*, Vol. 40, No. 5, 2008, pp. 1724 – 1731. doi: https://doi.org/10.1016/j.aap.2008.06.012.
20. Roland, J., P. Way, and M. Sartipi. Studying the Effects of Weather and Roadway Geometrics on Daily Accidents. *Proceedings of Cyber-Physical Systems and Internet-of-Things*, 2019.
21. Bisong, E. More supervised machine learning techniques with scikit-learn. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Springer, 2019, pp. 287–308.
22. Ramchoun, H., M. A. J. Idrissi, Y. Ghanou, and M. Ettaouil. Multilayer Perceptron: Architecture Optimization and Training. *IJIMAI*, Vol. 4, No. 1, 2016, pp. 26–30.
23. Ranjan, C. Extreme Rare Event Classification using Autoencoders in Keras. *Proceedings of Towards Data Science*.
24. Wilson, D. Using Machine Learning to Predict Car Accident Risk. *Proceedings of Medium - Geospatial Artificial Intelligence*.
25. Ziakopoulos, A. and G. Yannis. A review of spatial approaches in road safety. *Accident Analysis and Prevention*, Vol. 135, 2020, p. 105323. doi:https://doi.org/10.1016/j.aap.2019.105323.