Coping with Class Imbalance in Classification of Traffic Crash Severity based on Sensor and Road Data: A Feature Selection and Data Augmentation Approach

Deepti Lamba[1] Majed Alsadhan[1] William Hsu[1]
Eric Fitzsimmons[2] Gregory Newmark[3]

[1]Department of Computer Science [2]Department of Civil Engineering
[3]Department of Landscape Architecture and Regional & Community Planning
Kansas State University
{dlamba | mos | bhsu | fitzsimmons | gnewmark}@ksu.edu

ABSTRACT

This paper presents machine learning-based approaches to classification of historical traffic crashes in Kansas by severity, applied to a data set consisting of highway geometry, weather, and road sensor data. The goal of this work is to identify relevant features using a variety of loss measures and algorithms for feature selection. This is shown to facilitate the discovery of the most relevant sensors for the task of learning to predict severe crashes (those involving bodily injury). The key technical challenges are to cope with class imbalance (as a 75% majority of crashes are non-severe) and a highly correlated and redundant set of features from multiple coalesced sources. The major novel contributions of this work are the development of a random oversampling strategy for data augmentation, combined with the systematic application of multiple feature selection measures over a range of supervised inductive learning models and algorithms. Positive results from this approach, on a data set of 277 initial ground features and 20,000 vehicle crashes collected over 9 years (2007 – 2015) by the Kansas Department of Transportation (KDOT), included models trained using 30 features (out of 277) that achieve cross-validation precision and recall comparable to those obtained using the full set of features. These and other results point towards potential use of feature selection findings and the resultant models in planning future road construction.

KEYWORDS

Machine Learning, Class Imbalance, Predictive Analytics, Feature Selection, Data Augmentation, Traffic Engineering

1. INTRODUCTION

This work addresses the problem of applying machine learning to historical training data from roadside sensors, in conjunction with weather data and road geometry data, in order to predict the severity level of crashes based on new sensor and weather data. Roadside sensors are usually deployed along the highly concentrated highways to provide real-time information about the traffic including volume, speed of vehicles, and the vehicle count. Further details of the data set are explained below. The research aim discussed in this paper is to build a predictive model from offline data to identify risky factors and to test the hypothesis that sensor data, along with weather and road data, are adequate to make accurate, precise, and sensitive (high-recall) predictions of discrete severity level. The success of the model provides a use case and applied rationale for deployment of such sensors in a wide area and technical objectives for development and refinement of sensors that can capture weather data to get a precise estimation of weather at the point

of deployment.

Need and Significance: Every year traffic accidents hurt the economy by claiming numerous human lives and causing damage to public property. According to USDOT (United States Department Of Transportation) [1], 37,133 people lost their lives in road crashes in 2017 alone. An early estimate of fatalities for the first half of 2018 shows that an estimated 17,120 people lost their lives. These statistics suggest a need for better road safety mechanisms. Despite the number and significance of advances that have been made in the field of road safety, more efforts are needed to identify factors that lead to a severe crash so that measures can be taken to mitigate the identified risks.

One of the issues with such data is significant class imbalance between severe and non-severe classes [2]. If this data imbalance is not taken into account, then the classification model built may be biased and inaccurate. Standard classification algorithms have a bias towards the majority class. The features of the minority class are treated as noise and are often ignored in the model building process. This results in a model that achieves a significantly high accuracy by always predicting the majority class. To handle imbalance, we employ different sampling techniques: data-level techniques such as undersampling, oversampling, re-sampling with replacement (bootstrapping) and re-sampling without replacement. The KDOT data consists of crash severity coded as "PO" (Property Damage), "I" (Injury), and "F" (Fatal). For the purpose of severity prediction, the "I" and "F" classes have been combined to form the Severe class (constituting 24% of the examples) and "PO" is considered to be the Non-Severe class (constituting 76%). These traffic sensors yield a tremendous amount of data and its full potential is yet to be tapped.

The main contributions in this paper include:

- The development of a **coalesced, multisensor data set** consisting of ground sensor, weather, and road geometry data used to train a supervised learning system for classification of crashes by predicted severity.
- The application and evaluation of numerous **feature selection methods** to identify **the most relevant features for crash prediction** based on criteria such as validation set accuracy, prediction, and recall.
- The application of **data augmentation techniques** to cope with the existing imbalance in the data set.
- The application and evaluation of a range of **discriminative classifiers** (especially linear discriminants and decision tree-based models admitting rule-based explanation), along with ensemble methods.

We are interested in supervised inductive learning to classify the severity of historical crashes based on sensor, weather, and road geometry data but **without** using features from police reports or other *post hoc* (non-predictive) features. In this paper, a data set and test bed are presented that supports this learning task, starting with a survey of relevant machine learning methods for anomaly prediction in imbalanced data sets in section 2, data collection and preparation in the traffic analytics domain in section 3, and specific methods used for this severity prediction task in section 4, and continuing in section 5 with experimental findings on this task, interpreting and discussing these findings in section 6, and presenting conclusions and future work in section 7.

2. Literature Review

Extensive research has been done in crash severity prediction using several statistical, data mining, and machine learning approaches. These techniques have also been employed to identify crucial factors that lead to a severe crash. In [3], support vector machines (SVM) and ordered probit models were used for crash severity analysis; they showed that SVM produced better results than ordered probit models. In [4], SVMs were also used for severity prediction. The authors

used classification and regression trees (CART) [5] to identify important features and then train a model using SVM with polynomial and Gaussian radial basis kernels. This study showed that polynomial kernel outperforms the Gaussian radial basis kernel.

In [6], crash severity was analyzed using crash reports, real-time traffic, and weather data using random forests to rank the variables by importance. The authors used SVM and logit models for severity prediction. This study demonstrated that weather data and real-time traffic variables are significant factors in severity prediction. In [7], real-time traffic and weather data was also used to predict severity and occurrence of crashes. The authors of this work also used random forests to rank the features according to importance and then build models using Bayesian logistic regression and logit models. This study concluded that weather parameters did not have a direct influence on accident severity.

In [8], several techniques for handling class imbalance such as undersampling, oversampling, and ensemble methods (majority voting and bagging) were used. The authors also experimented with several classification algorithms such as logistic regression, decision trees, neural networks, gradient boosting models, and Naïve Bayes classifiers. For these experiments and this task, over-sampling with random forests produces the best classification performance. In [9], a comparison was conducted between four statistical and machine learning methods such as multinomial logit (MNL), nearest neighbor classification (NNC), support vector machines (SVM) and random forest (RF). This study showed that nearest neighbor classification gives the best overall result.

In [10], the simultaneous influence of human factors, road, vehicle, weather conditions and traffic features were explored in crash severity prediction. The authors used a series of artificial neural networks to model crash severity and to identify significant factors that lead to a severe crash. In [11], a CART model was used for injury severity and their results identified vehicle type as the most important feature associated with crash severity.

3. Data Collection and Processing

The data set used in this paper covers segments of the Kansas City highway network that are monitored by sensors, and traffic monitors which fall within the state of Kansas. This data set is a fusion of data from four different sources:

- Crash Data
- Road Geometry Data
- Traffic Sensor Data
- Weather Data

3.1. Crash Data

Crash data consists of all the crashes that occurred within 500 meters upstream/downstream of a traffic sensor. The data were collected over a period of nine years from the beginning of year 2007 up to the end of 2015. These data were derived from police reports that were completed by a highway patrol officers at the scene of a crash, and were obtained from the Kansas Crash and Analysis Reporting System (KCARS) database of the Kansas Department of Transportation (KDOT) [1]. There were a total of 19,881 crashes during the nine year period. The crash severity was coded as: "PO" (Property Damage), "F" (Fatal) and "I" (Injury). Table 1 shows the distribution of crashes between the severe class, which is formed by merging fatal and injury classes, and the non-severe class which is a property damage class.

Since this study focuses on data obtained from road geometry, weather data, and traffic sensors, we only extract four attributes from the crash data obtained from police reports: time/date of accidents, days of accidents, locations of accidents (in terms of latitudes and longitudes), and

severity of the crashes. Latitude and longitude coordinates are not used as part of analysis, but are required for the merger of this data with road geometry data. The combination of time/date and location were used to further merge the data with Traffic Sensors data. The time/date of accidents were used to merge the data with weather data.

Severity Type	#Crashes
Severe (F and I)	4,771 (23.99%)
Non-Severe (PO)	15,110 (76.00%)

Table 1: Number of crashes based on severity

3.2. Road Geometry Data

The road geometry data were provided by Kansas Department Of Transportation (KDOT) for the nine-year period. This data includes fields: route direction, number of lanes, width of lanes, medians, median barrier type, shoulders and information on elevation, which was further used to calculate slope. Poly-line data were used to calculate curve radii (in degrees).

3.3. Traffic Sensor Data

This data were obtained from the Kansas City (KC) Scout Project. The KC Scout Project is responsible for monitoring more than 300 miles of highway in the region by means of sensors, which are mounted along the roadways, and video cameras. The sensors are used to record the speed of traffic, count of vehicles, and volume of traffic. The time/date stamps and location from the crash data were used to obtain the traffic sensor data for the quarter hour before and after any crash. These data were obtained based on the proximity of sensors to the accident site.

3.4. Weather Data

The weather data were extracted from the National Oceanic and Atmospheric Administration (NOAA) National Centers for Environmental Information Surface Data. These records are collected every hour at the downtown Kansas City airport, which is located 15 miles north to the center of the area used in this study. The weather data are not considered very precise due to the distance from the accident area to the airport location. However, they give fairly correct estimates of weather condition at the time of the crash.

3.5. Data Cleaning

Records with more than 20 missing values were manually removed from the data set. The number of crashes reduced to 19,881 from 20,822. A distribution of crashes based on severity is shown in Table 1. The categorical data were coded with numerical values. Location features (those based **only** on a GPS coordinate such as latitude and longitude) were removed from the data set because these are not generalizable based on the spatial data model of this data set. (For example, the sensors are not themselves georeferenced, nor are they placed according to any uniform or regular scheme.) Features from crash data were removed as well (except for the 3 features mentioned in section 3.1) This reduced the number of features from 277 to 133 features. The final numbers of crashes and features are shown in Table 2.

4. METHODOLOGY

In this paper, six classification models were used for training the data. This section begins by providing details of the feature selection methods used in this paper. These details are followed

Data	Number
Features	133
Instances	19,881

Table 2: Number of Features and Instances

by a discussion of data augmentation methods used for handling the problem of class imbalance. Afterwards, a summary of the classification models that were used for training is given, followed by a discussion of evaluation metrics used in the paper. Finally, this section provides a few details about the tools used for this paper along with the experimental setup.

4.1. Feature Selection

Feature selection is the process of finding the most important features/input variables. It usually results in a reduction of number of the input variables or features that will be used for building a machine learning model [12]. Feature selection has two major benefits:

- It eliminates irrelevant or redundant data that would otherwise makes it more difficult to discover meaningful patterns in the data. Hence, identifying those features that will build a better quality model.
- If the data is high dimensional then most machine learning algorithms require a larger data set for training. It also requires more computational resources.

Feature selection methods are broadly classified into two categories: filter and wrapper methods [13]. Filter methods [14] are independent of any prediction algorithm and rely on the general characteristics of the data. Wrapper methods, on the other hand, rely on classification algorithm to evaluate subsets of features [15].

4.1.1. Information Gain Attribute Ranking

This method evaluates the quality of an attribute by calculating the information gain (i.e., the change in entropy due to conditioning) for each attribute with respect to the output variable (class) [16] [17] [18] as shown in equation 1.

$$IG(C, A) = H(C) - H(C \mid A) \tag{1}$$

where, H is the information entropy, C is the class, and A is the attribute.

4.1.2. Gain Ratio Attribute Evaluation

This method evaluates the quality of an attribute by measuring the gain ratio (GR) with respect to the class [18]. Equation 2 shows the formula for gain ratio attribute evaluation.

$$GR(C, A) = \frac{H(C) - H(C \mid A)}{H(A)} \tag{2}$$

where, C is the class, H is the information entropy, and A is the attribute.

4.1.3. Correlation-Based Attribute Evaluation (Pearson's Coefficient)

This method measures the correlation between each independent and dependent variable. A negative value indicates negative correlation between two variables, whereas a positive value signifies a positive correlation, and a value of 0 indicates the absence of a relationship between variables. The Pearson correlation [19] is measured in equation 3.

$$\rho = \frac{\text{cov}(X, Y)}{\sigma_x \sigma_y} \tag{3}$$

where, cov designates the co-variance, X the input feature, Y the output feature, σ_x is the standard deviation of X, and σ_y is the standard deviation of Y.

4.1.4. Chi-Squared Attribute Evaluation

Compute the chi-squared statistic of each attribute with respect to the class [18]. Chi-square [20] is measured using equation 4 where a high value signifies that two features are dependent.

$$\tilde{\chi}^2 = \frac{1}{d} \sum_{k=1}^{n} \frac{(O_k - E_k)^2}{E_k} \tag{4}$$

where: K is the number of classes, O_k = observed value, n = total number of classes in data-set, and E_k = expected value

4.1.5. Correlation-Based Feature Subset Evaluation

This objective of this method is to evaluate the feature-class and feature-feature correlation [21]. Subsets with the highest merit found during the search are selected using equation 5.

$$M_s = \frac{k \cdot \overline{r_{cf}}}{\sqrt{k + k(k-1) \cdot \overline{r_{ff}}}} \tag{5}$$

where, M_s is heuristic of a feature subset s, K is the number of features, r_{cf} is the mean feature-class correlation, and r_{ff} is the average feature-feature inter-correlation.

4.1.6. Wrapper Subset Evaluation

The main idea of wrapper subset evaluation [15] is to find a good subset using the validation-set accuracy of a supervised inductive learning algorithm (inducer) as part of the evaluation function. This method results in better features as compared to filter methods because it is tied to a specific inducer.

4.2. Data Augmentation

4.2.1. Data-Level Resampling

Data-level approaches work by re-balancing the class distribution [22]. These include oversampling and undersampling techniques. Oversampling techniques create more of the minority class whereas undersampling techniques remove some of the majority class. We experimented with the following oversampling techniques: random oversampling, SMOTE [23], borderline SMOTE [24], SVM SMOTE [25] and Adaptive Synthetic (ADASYN) [26]. Undersampling techniques used in our preliminary experiments include: random undersampling [27], Tomek's links [28], edited nearest neighbors [29], condensed nearest neighbor [30], one-sided selection [31], and neighborhood cleaning rules [32].

4.2.2. Algorithmic Resampling

Algorithmic methods include bootstrap sampling with or without replacement, which produces a single random subsample of the data [33] on which a supervised learning classifier is then trained. This is distinguished from data-level resampling in that the bootstrap sample distribution uniformly draws from the original instances, maintaining the relative frequency of labels.

4.2.3. Ensemble Methods

Ensemble methods train several classifiers on training data and their evaluations are aggregated to produce the final classification decision [34].

4.3. Classification Models

This paper uses the following classification algorithms to build models:

- Logistic Regression (LR) [35]
- Naïve Bayes (NB) [36]
- Support Vector Machines (SVM) [37]
- Decision Trees: CART [5]
- Random Forest (RF) [38]
- Extremely Randomized Trees (ERT) [39]

4.4. Experimental Setup and Evaluation Metrics

Weka [40] is used for feature selection, 10-fold cross validation was applied for validating the classification models [41]. In 10-fold cross validation, data is partitioned into 10 randomly selected folds (subsets) of roughly equal size: nine folds are used for training, and the remaining one fold is used for validation. This process is repeated 10 times such that each subset is used exactly once for validation.

Feature selection methods are run to produce the best 10, 15, 20, 25, and 30 features for each of the methods. These feature sets are then used for classification using the algorithms mentioned in section 4.3. Based on the results we selected 30 features for each feature set. We tested our selected features with different sampling techniques mentioned in section 4.2. We run over 5000 experiments using these feature sets with sampling techniques. These experiments were done using the feature sets and exposing them to sampling techniques by running classification algorithms to produce the results. The classification models were built using python library `sklearn`. Support vector machines was tested using linear, polynomial, sigmoid and rbf kernels. In our experiments, SVM performed best with rbf (radial basis function) kernel.

The performance of the classifiers is evaluated using a confusion matrix.This enables us to calculate accuracy, precision sensitivity (recall) and f-score [42] Since we are using 10 fold cross-validation, we will present the scores as the mean of all the 10 folds. So we will express our results in terms of weighted accuracy, weighted precision, weighted recall and weighted f-score.

4.5. Baselines

We employ two classification models as our baselines:

- Original Data (O.D): The original data with 133 features with logistic regression as the inductive learning classifier gives us the highest weighted accuracy. This is the scenario where the model classifies everything as the majority class.
- Original Data (O.D) + Data Augmentation: The original data with 133 features is augmented using different sampling techniques and it performs best with random oversampling and SVM to give a weighted accuracy of 96.62%.

5. Experimental Results

Table 3 shows the results of six classifiers used with four feature sets consisting of 30 features each. Info gain, gain ratio, and chi squared feature selection methods produced the same attributes; hence, the same results for accuracy, precision, recall and F-score. So henceforth, we show only Info gain in the results. Wrapper subset evaluation was used with random forest for feature selection and achieves the best weighted accuracy of 75.85% when compared to to other feature sets with random forest classifier. We can observe from Table 3 that feature selection alone does significantly improve the weighted precision over the baseline for logistic regression model.

FS	Algo	Wt. Accuracy(%)	Wt. Precision(%)	Wt. Recall(%)	Wt. F-score(%)
Info Gain	Logistic Regression	57.77 ± 1.37	**76.00** ± 0.90	65.64 ± 1.22	-
	Naïve Bayes	67.30 ± 7.63	65.03 ± 1.43	67.30 ± 7.63	64.59 ± 5.15
	Random Forest	**75.50** ± 0.88	65.92 ± 2.31	**75.50** ± 0.88	**66.33** ± 1.24
	SVM	65.92 ± 2.31	75.50 ± 0.88	66.33 ± 1.24	-
	Extreme Tree	64.15 ± 0.99	63.98 ± 1.22	64.15 ± 0.99	64.05 ± 1.04
	CART	63.09 ± 0.61	64.07 ± 0.87	63.09 ± 0.61	63.55 ± 0.65
Pearson Corr.	Logistic Regression	58.52± 3.23	**75.97** ± 0.91	65.63 ± 1.24	-
	Naïve Bayes	68.85 ± 7.35	65.18 ± 0.85	68.85 ± 7.35	65.17 ± 4.61
	Random Forest	**75.51** ± 0.91	65.53 ± 2.51	**75.51** ± 0.91	**66.21** ± 1.24
	SVM	65.53 ± 2.51	75.51 ± 0.91	66.21 ± 1.24	-
	Extreme Tree	63.75 ± 0.91	63.80 ± 1.34	63.75 ± 0.91	63.77 ± 1.09
	CART	63.54 ± 1.27	64.35 ± 1.44	63.54 ± 1.27	63.92 ± 1.28
CFS	Logistic Regression	57.77 ± 1.37	**75.98** ± 0.91	65.63 ± 1.22	-
	Naïve Bayes	61.99 ± 8.00	65.16 ± 1.62	61.99 ± 8.00	61.94 ± 6.25
	Random Forest	**75.52** ± 0.96	66.38 ± 2.55	**75.52** ± 0.96	**66.95** ± 1.27
	SVM	66.38 ± 2.55	75.52 ± 0.96	66.45 ± 1.27	-
	Extreme Tree	63.81 ± 1.21	63.91 ± 1.50	63.81 ± 1.21	63.85 ± 1.33
	CART	63.40 ± 0.88	63.93 ± 0.77	63.40 ± 0.88	63.65 ± 0.70
Wrapper	Logistic Regression	57.77 ± 1.37	**75.99** ± 0.90	65.64 ± 1.22	-
	Naïve Bayes	38.65 ± 11.40	65.86 ± 1.61	38.65 ± 11.40	36.22 ± 10.44
	Random Forest	**75.85** ± 0.94	61.19 ± 4.39	**75.85** ± 0.94	65.62 ± 1.26
	SVM	61.19 ± 4.39	75.85 ± 0.94	65.62 ± 1.26	-
	Extreme Tree	73.15 ± 1.09	63.60 ± 1.58	73.15 ± 1.09	66.19 ± 1.35
	CART	72.47 ± 0.81	63.61 ± 1.15	72.47 ± 0.81	**66.20** ± 1.13
Original data (Baseline)	Logistic Regression	**76.00** ± 0.88	63.92 ± 9.44	**76.00** ± 0.88	65.68 ± 1.17
	Naïve Bayes	65.01 ± 6.88	64.45 ± 1.03	65.01 ± 6.80	63.66 ± 4.43
	Random Forest	75.00 ± 0.83	65.09 ± 2.07	75.00 ± 0.83	**66.41** ± 1.25
	SVM	75.72 ± 0.87	**66.43** ± 2.84	75.72 ± 0.87	66.13 ± 1.13
	Extreme Tree	63.64 ± 1.18	64.48 ± 1.38	63.64 ± 1.18	64.04 ± 1.25
	CART	62.91 ± 1.00	64.11 ± 1.09	62.91 ± 1.00	63.48 ± 0.98

Table 3: Results(%) of supervised inductive learning classifiers with feature selection methods (including original baseline)

Next we expose these data sets to different data augmentation methods mentioned in section 4.2. Random oversampling produced the best results, followed by re-sampling with replacement (bootstrap aggregation). Random undersampling produced the best results as compared to other undersampling methods. Table 4 shows the results of classification with support vector machines for all the feature sets. We present only the results for support vector machines because it produced the best results as compared to other methods: logistic regression, naïve bayes, decision tree (CART), extremely randomized tree and random forest.

The results demonstrate that info gain feature set gives slightly better results than the other feature selection methods in terms of precision, recall and f-score. Our baseline when treated with data augmentation methods produces a weighted accuracy of 96.62% for random over sampling. The results obtained using a subset of 30 out of 133 features are comparable to those obtained using baseline of all 133 features.

Thus, feature selection methods other than the wrapper method produce comparable results to our second baseline (i.e. original data + data augmentation), and help in identifying the 30 most important features that produce these results. Table 5 shows these 30 features and they are ranked using random forest with the most important feature at the top and least important being at the bottom.

FS Method	Evaluation Measure	Random Over Sampling	Random Under Sampling	Resample with replace	Resample no replace
Info Gain	Wtd. Acc.	**95.40** ± 0.31	49.69 ± 0.94	89.16 ± 0.55	75.52 ± 0.42
	Wtd. Precision	**95.49** ± 0.29	53.52 ± 4.17	89.93 ± 0.51	66.47 ± 1.79
	Wtd. Recall	**95.40** ± 0.31	49.69 ± 0.94	89.16 ± 0.55	75.52 ± 0.42
	Wtd. F-score	**95.40** ± 0.31	36.39 ± 1.07	88.12 ± 0.65	66.37 ± 0.49
Pearson Corr.	Wtd. Acc.	**95.38** ± 0.39	49.70 ± 1.24	89.19 ± 0.50	75.58 ± 0.47
	Wtd. Precision	**95.45** ± 0.37	52.49 ± 2.44	89.99 ± 0.45	66.71 ± 2.14
	Wtd. Recall	**95.38** ± 0.39	49.70 ± 1.24	89.19 ± 0.50	75.58 ± 0.47
	Wtd. F-score	**95.38** ± 0.39	37.37 ± 1.74	88.15 ± 0.59	66.33 ± 0.55
CFS	Wtd. Acc.	**94.76** ± 0.43	51.48 ± 1.73	88.93 ± 0.60	75.52 ± 0.42
	Wtd. Precision	**94.81** ± 0.43	54.53 ± 1.79	89.69 ± 0.58	66.53 ± 1.30
	Wtd. Recall	**94.76** ± 0.43	51.48 ± 1.73	88.93 ± 0.60	75.52 ± 0.42
	Wtd. F-score	**94.75** ± 0.43	41.71 ± 2.52	87.85 ± 0.70	66.44 ± 0.52
Wrapper	Wtd. Acc.	56.71 ± 0.99	51.66 ± 1.65	**76.36** ± 0.76	75.87 ± 0.43
	Wtd. Precision	56.78 ± 0.98	51.73 ± 1.63	**75.66** ± 1.85	61.73 ± 5.27
	Wtd. Recall	56.71 ± 0.99	51.66 ± 1.65	**76.36** ± 0.76	75.87 ± 0.43
	Wtd. F-score	56.62 ± 1.04	51.64 ± 1.66	**67.06** ± 1.06	65.64 ± 0.59
Baseline	Wtd. Acc.	**96.62** ± 0.26	49.04 ± 1.10	89.65 ± 0.46	75.76 ± 0.44
	Wtd. Precision	**96.73** ± 0.24	52.34 ± 6.86	90.61 ± 0.40	67.03 ± 2.10
	Wtd. Recall	**96.62** ± 0.26	49.04 ± 1.10	89.65 ± 0.46	75.76 ± 0.44
	Wtd. F-score	**96.62** ± 0.26	33.61 ± 1.33	88.65 ± 0.55	66.21 ± 0.57

Table 4: Results(%) of SVM with Feature Selection and Data Augmentation Methods

Attribute Name	Description	Source of Data
HCUR_SOURCE	Horizontal curve data	Road Geometry Data
MEDN_WDTH	Median width	Road Geometry Data
KC_TEMP+30	Temp. 30 min. after crash	Weather Data
KC_TEMP+15	Temp. 15 min. after crash	Weather Data
KC_TEMP:0	Temp. 15 min. at crash	Weather Data
KC_TEMP-15	Temp. 15 min. before crash	Weather Data
Station_Speed:0	Average Speed of vehicles	Traffic Sensor Data
KC_DEWP-15	Dew point 15 min. before crash	Weather Data
KC_DEWP:0	Dew point at crash	Weather Data
KC_DEWP+15	Dew point 15 min. after crash	Weather Data
KC_DEWP+30	Dew point 30 min. after crash	Weather Data
Station_Volume+30	Station Count changed to 1 hr. volume	Traffic Sensor Data
MED_TYPE_Desc	Description of median type	Road Geometry Data
Station_Count+30	Vehicle count 30min. after crash	Traffic Sensor Data
DAY_OF_ACCIDENT	Day of crash	Crash Data
Station_Volume+15	Station Count changed to 1 hr. volume	Traffic Sensor Data
Station_Count+15	Vehicle count 15min. after crash	Traffic Sensor Data
KC_DIR:0	Wind direction at crash	Weather Data
Station_Speed_Comp+30	Range of Data completeness	Traffic Sensor Data
Station_Speed-15	Average Speed 15 min before crash	Traffic Sensor Data
Station_Speed_Comp+15	Range of Data completeness	Traffic Sensor Data
KC_DIR-15	Wind dir. 15 min. before crash	Weather Data
IsPeakHourAM+30	30 min. after peak AM hour	Traffic Sensor Data
KC_VSB+30	Visibility 30 min. after crash	Weather Data
KC_VSB+15	Visibility 15 min. after crash	Weather Data
IsPeakHourAM:0	During peak AM hour	Traffic Sensor Data
IsPeakHourAM+15	15 min after peak AM hour	Traffic Sensor Data
KC_GUS+15	Wind gust 15 min. after crash	Weather Data
KC_GUS+30	Wind gust 30 min. after crash	Weather Data
KC_GUS:0	Wind gust at crash	Weather Data

Table 5: Features selected using information gain are ranked by random forest classifier

We use the features selected by information gain (info gain), which are presented in Table 5 for building classification models.

Table 6 shows the class-wise results of applying random oversampling to info gain feature set and building our six classification models. We clearly see a 25.52% increase in performance over our O.D. baseline and a comparable performance to our O.D. + random oversampling baseline.

Algorithm	Class	Random O.S			O.D.			O.D. + Random O.S		
		Pr.	Recall	F-score	Pr.	Recall	F-score	Pr.	Recall	F-score
Random Forest	Severe	92	**95**	93	30	03	06	95	94	**95**
	Non-Severe	94	92	93	76	**98**	86	94	95	**95**
	Wtd. Avg.	93	93	93	65	75	66	95	95	**95**
Extreme Tree	Severe	78	**95**	**86**	26	28	27	**95**	74	83
	Non-Severe	**94**	73	82	77	75	76	79	**96**	**87**
	Wtd. Avg.	86	84	84	64	64	64	**87**	**85**	**85**
CART	Severe	78	95	**85**	25	28	26	**95**	74	83
	Non-Severe	**94**	73	82	76	74	75	79	**96**	**87**
	Wtd. Avg.	86	84	84	64	63	63	**87**	**85**	**85**
SVM	Severe	**97**	93	95	36	01	03	94	**99**	**97**
	Non-Severe	94	**97**	95	76	99	86	**99**	94	**97**
	Wtd. Avg.	95	95	95	66	76	66	**97**	**97**	**97**
LR	Severe	53	**55**	**54**	31	00	00	**54**	54	**54**
	Non-Severe	**54**	52	**53**	76	**100**	86	**54**	53	**53**
	Wtd. Avg.	53	53	53	**65**	**76**	**66**	54	54	54
Naïve Bayes	Severe	**53**	52	53	26	24	25	51	**70**	**59**
	Non-Severe	53	54	53	**77**	**78**	**77**	53	34	41
	Wtd. Avg.	53	53	53	**64**	**65**	**65**	52	52	50

Table 6: Results(%) of binary classification with random over sampling for info gain feature set vs the two baselines

6. Discussion

Information gain-based feature selection selected the features given in Table 5 that have been ranked using a random forest classifier. Out of the 30 features, 11 come from roadside sensor data, 15 from weather data and 3 from road geometry data. Results in Table 4 and Table 6 show that features selected using information gain when subjected to random over sampling produce the best results with SVM, followed by random forests. The augmented data also produced better results than our baseline of original data when used with tree-based methods (CART and extremely randomized trees). We get an increase of 25.5% in accuracy, 49.4% in precision, and 25.5% in recall when our method (info gain feature set with random-oversampling using SVM) is compared to original data baseline. We conducted a paired T-test between our method and our second baseline with SVM classifier for 10-fold cross validation, resulting in a p-value of 0.000000022 (2.2×10^{-8}) at the 95% level of confidence.

7. Conclusions and Future Work

We have presented machine learning-based approaches to classify historical traffic crashes in Kansas by severity. This work resulted in identifying relevant features using feature selection algorithms. The key technical challenges faced were simultaneous class imbalance and a high number of correlated, weakly-relevant features. Our experimental findings demonstrate empirically, with high statistical significance, that precision and recall of severity prediction are improved by algorithmic feature selection and random oversampling, and that thereby only a subset of 30 features (out of over 133) is needed.

By systematically comparing six feature selection methods across six supervised inductive models (all discriminative except Naïve Bayes), and applying multiple oversampling techniques to the best-performing of these combinations (those with highest precision and recall), we conclude that

feature selection using information gain with random oversampling produces a cross-validation precision and recall higher than those obtained using the full set of features without any data augmentation. We also achieved results of comparable quality to the full 133-feature set using feature selection with our sampling-based data augmentation approach, which reduced the feature subset size to 30. These results point towards potential use of feature selection findings and the resultant models in planning future road construction.

In current and future work, we are planning to explore more ensemble methods and cost-sensitive methods for the traffic crash severity prediction task. Some of these methods may also facilitate transfer learning to address related **tasks** such as traffic crash prediction (by risk, location, and road condition) and related **domains** such as crash severity in other geographic areas, with different sensors or weather data (and forecasting models), and with different roads and commensurate traffic density (e.g., rural areas).

8. ACKNOWLEDGMENTS

We would like to thank the following members of Laboratory for Knowledge Discovery in Databases (KDD) at Kansas state University for processing the initial raw data and building the database: Mary Grace Blair, Luis Enrique Bobadilla, Jeffrey Cook, Sandeep Dasari, Ray Luo, and Yihong Theis.

9. REFERENCES

[1] US Department of Transportation, "US Department of Transportation," 2019.

[2] C. Drummond and R. C. Holte, "Explicitly representing expected cost: An alternative to roc representation," in *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 198–207, ACM, 2000.

[3] Z. Li, P. Liu, W. Wang, and C. Xu, "Using support vector machine models for crash injury severity analysis," *Accident Analysis & Prevention*, vol. 45, pp. 478–486, 2012.

[4] C. Chen, G. Zhang, Z. Qian, R. A. Tarefder, and Z. Tian, "Investigating driver injury severity patterns in rollover crashes using support vector machine models," *Accident Analysis & Prevention*, vol. 90, pp. 128–139, 2016.

[5] L. Breiman, J. Friedman, R. Olshen, and C. Stone, "Classification and regression trees. wadsworth int," *Group*, vol. 37, no. 15, pp. 237–251, 1984.

[6] R. Yu and M. Abdel-Aty, "Analyzing crash injury severity for a mountainous freeway incorporating real-time traffic and weather data," *Safety science*, vol. 63, pp. 50–56, 2014.

[7] A. Theofilatos, "Incorporating real-time traffic and weather data to explore road accident likelihood and severity in urban arterials," *Journal of safety research*, vol. 61, pp. 9–21, 2017.

[8] H. Jeong, Y. Jang, P. J. Bowman, and N. Masoud, "Classification of motor vehicle crash injury severity: A hybrid approach for imbalanced data," *Accident Analysis & Prevention*, vol. 120, pp. 250–261, 2018.

[9] A. Iranitalab and A. Khattak, "Comparison of four statistical and machine learning methods for crash severity prediction," *Accident Analysis & Prevention*, vol. 108, pp. 27–36, 2017.

[10] F. R. Moghaddam, S. Afandizadeh, and M. Ziyadi, "Prediction of accident severity using artificial neural networks," *International Journal of Civil Engineering*, vol. 9, no. 1, p. 41, 2011.

[11] L.-Y. Chang and H.-W. Wang, "Analysis of traffic injury severity: An application of non-parametric classification tree techniques," *Accident Analysis & Prevention*, vol. 38, no. 5, pp. 1019–1027, 2006.

[12] E. M. Karabulut, S. A. Özel, and T. İbrikçi, "A comparative study on the effect of feature selection on classification accuracy," *Procedia Technology*, vol. 1, pp. 323 – 327, 2012. First

World Conference on Innovation and Computer Sciences (INSODE 2011).
[13] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.
[14] N. Sánchez-Maroño, A. Alonso-Betanzos, and M. a. Tombilla-Sanromán, "Filter methods for feature selection–a comparative study," in *International Conference on Intelligent Data Engineering and Automated Learning*, pp. 178–187, Springer, 2007.
[15] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.
[16] M. A. Hall and G. Holmes, "Benchmarking attribute selection techniques for discrete class data mining," 2002.
[17] J. Novakovic, "Using information gain attribute evaluation to classify sonar targets," in *17th Telecommunications forum TELFOR*, pp. 1351–1354, 2009.
[18] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.
[19] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of machine learning research*, vol. 3, no. Mar, pp. 1157–1182, 2003.
[20] H. Liu and R. Setiono, "Chi2: Feature selection and discretization of numeric attributes," in *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*, pp. 388–391, IEEE, 1995.
[21] M. A. Hall, "Correlation-based feature selection for machine learning," 1999.
[22] A. Ali, S. M. Shamsuddin, and A. L. Ralescu, "Classification with class imbalance problem: a review," *Int. J. Advance Soft Compu. Appl*, vol. 7, no. 3, pp. 176–204, 2015.
[23] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.
[24] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-smote: a new over-sampling method in imbalanced data sets learning," in *International conference on intelligent computing*, pp. 878–887, Springer, 2005.
[25] H. M. Nguyen, E. W. Cooper, and K. Kamei, "Borderline over-sampling for imbalanced data classification," in *Proceedings: Fifth International Workshop on Computational Intelligence & Applications*, vol. 2009, pp. 24–29, IEEE SMC Hiroshima Chapter, 2009.
[26] H. He, Y. Bai, E. A. Garcia, and S. Li, "Adasyn: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 1322–1328, IEEE, 2008.
[27] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–449, 2002.
[28] I. Tomek, "Two modifications of cnn," *IEEE Trans. Systems, Man and Cybernetics*, vol. 6, pp. 769–772, 1976.
[29] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 3, pp. 408–421, 1972.
[30] P. Hart, "The condensed nearest neighbor rule (corresp.)," *IEEE transactions on information theory*, vol. 14, no. 3, pp. 515–516, 1968.
[31] M. Kubat, S. Matwin, *et al.*, "Addressing the curse of imbalanced training sets: one-sided selection," in *Icml*, vol. 97, pp. 179–186, Nashville, USA, 1997.
[32] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," in *Conference on Artificial Intelligence in Medicine in Europe*, pp. 63–66, Springer, 2001.
[33] D. Basu, "On sampling with and without replacement," *Sankhyā: The Indian Journal of Statistics*, pp. 287–294, 1958.
[34] M. Hamza and D. Larocque, "An empirical comparison of ensemble methods based on clas-

sification trees," *Journal of Statistical Computation and Simulation*, vol. 75, no. 8, pp. 629–643, 2005.
[35] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.
[36] H. Zhang, "The optimality of naive bayes," *AA*, vol. 1, no. 2, p. 3, 2004.
[37] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
[38] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
[39] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, pp. 3–42, Apr. 2006.
[40] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *SIGKDD Explorations*, vol. 11, no. 1, pp. 10–18, 2009.
[41] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, vol. 14, pp. 1137–1145, Montreal, Canada, 1995.
[42] R. Parikh, A. Mathai, S. Parikh, G. C. Sekhar, and R. Thomas, "Understanding and using sensitivity, specificity and predictive values," *Indian journal of ophthalmology*, vol. 56, no. 1, p. 45, 2008.