sustainability

MDPI

Article

Analysis of Factors Affecting Real-Time Ridesharing Vehicle Crash Severity

Bei Zhou [1], Xinfen Zhang [1,*], Shengrui Zhang [1], Zongzhi Li [2] and Xin Liu [1]

[1] School of Highway, Chang'an University, Xi'an 710064, China; bzhou3@chd.edu.cn (B.Z.); zhangsr@chd.edu.cn (S.Z.); lxin@chd.edu.cn (X.L.)

[2] Department of Civil, Architectural, and Environmental Engineering, Illinois Institute of Technology, Chicago, IL 60616, USA; lizz@iit.edu

* Correspondence: zxinfen@chd.edu.cn; Tel.: +86-187-9250-4794

Received: 28 May 2019; Accepted: 15 June 2019; Published: 17 June 2019

Abstract: The popular real-time ridesharing service has promoted social and environmental sustainability in various ways. Meanwhile, it also brings some traffic safety concerns. This paper aims to analyze factors affecting real-time ridesharing vehicle crash severity based on the classification and regression tree (CART) model. The Chicago police-reported crash data from January to December 2018 is collected. Crash severity in the original dataset is highly imbalanced: only 60 out of 2624 crashes are severe injury crashes. To fix the data imbalance problem, a hybrid data preprocessing approach which combines the over- and under-sampling is applied. Model results indicate that, by resampling the crash data, the successfully predicted severe crashes are increased from 0 to 40. Besides, the G-mean is increased from 0% to 73%, and the AUC (area under the receiver operating characteristics curve) is increased from 0.73 to 0.82. The classification tree reveals that following variables are the primary indicators of real-time ridesharing vehicle crash severity: pedestrian/pedalcyclist involvement, number of passengers, weather condition, trafficway type, vehicle manufacture year, traffic control device, driver gender, lighting condition, vehicle type, driver age and crash time. The current study could provide some valuable insights for the sustainable development of real-time ridesharing services and urban transportation.

Keywords: real-time ridesharing; crash severity; data imbalance; SMOTE+ENN; decision tree

1. Introduction

In recent years, with the rapid spread of GPS-enabled smartphones and communication technologies, real-time/dynamic ridesharing has become an important transportation mode which is transforming the urban mobility by providing convenient and timely transportation service. Unlike traditional ridesharing, real-time ridesharing provides substantial flexibility to both drivers and passengers by allowing them to arrange trips on short notice. Real-time ridesharing is defined by Amey et al. [1] as "A single, or recurring rideshare trip with no fixed schedule, organized on a one-time basis, with matching of participants occurring as little as a few minutes before departure or as far in advance as the evening before a trip is scheduled to take place". By the end of 2014, the leading ridesharing companies, such as Uber and Lyft, had penetrated 80% of U.S. cities with a population of at least 100,000 [2]. Previous studies have demonstrated that real-time ridesharing could promote the social and environmental sustainability in various ways, such as saving travel time and cost, reducing traffic congestions, and mitigating air pollution and so on [1,3–5]. Nevertheless, the booming market of real-time ridesharing has also brought some concerns, which could potentially hinder its development. One of the greatest concerns regarding this relatively new transportation mode is the potential traffic crash risks. For instance, a recent report has argued that the development of real-time

ridesharing is associated with a 2–4% increase in the number of fatal crashes [2]. In order to properly regulate the development of real-time ridesharing and fully utilize its potential benefits on urban transportation sustainability, it is critical to thoroughly investigate factors affecting crash severity of real-time ridesharing vehicles.

In the past decade, widespread real-time ridesharing has attracted attentions from various research fields. For instance, Amey et al. [1] discussed the potential benefits of real-time ridesharing. They also pointed out that the development of real-time ridesharing needed to overcome a series of technology, economic, behavioral, and institutional obstacles. Furuhata et al. [3] described the state of the art of the ridesharing system, including the classification of ridesharing system, ridesharing matching agencies, and challenges faced by matching agencies. Amirkiaee et al. [6] conducted a scenario-based survey to investigate factors affecting people's intentions to participate in real-time ridesharing. Ma et al. [5] studied the traffic flow patterns with different real-time ridesharing charges during the morning commute. Moreover, the optimization of ride-matching algorithms has also attracted considerable attentions [7–11]. However, to the best of the authors' knowledge, there is somewhat limited literature exploring factors affecting real-time ridesharing vehicle crash severity, which has motivated the current study to conduct a thorough analysis.

Although few studies have been carried out to specifically address traffic crashes regarding real-time ridesharing vehicles, significant research has been done to analyze crash severities from various aspects. Savolainen et al. [12] reviewed the evolution of research related to the statistical analysis of crash injury severities. They also discussed possible future methodology directions. To account for the unobserved characteristics of traffic crash data, Anastasopoulos et al. [13] proposed a multivariate tobit model to analyze crash injury severities. Yu and Abdel-Aty [14] developed three different models to analyze the injury severity for a mountainous freeway, including fixed parameter logit model, support vector machine, and random parameter logit model. The results demonstrated the substantial influences of real-time weather and traffic data on crash injury severity. To understand the contributing factors of rear-end crashes, Chen et al. [15] proposed a hybrid approach which combined the multinomial logit model and Bayesian network method. As vulnerable road users, the safety of pedestrians is always of particular concern. Haleem et al. [16] used the mixed logit model to compare factors affecting pedestrian crash severity at signalized and unsignalized intersections. Based on random parameter ordinal and multinomial regression models, Naik et al. [17] investigated injury severity of single-vehicle truck under various weather conditions. Zeng et al. [18] developed a nonlinear model-based mixed multinomial logit (MNL) approach to analyze factors contributing to crash severity. The results suggested that the proposed approach outperformed the standard mixed MNL model. To investigate factors affecting bicyclist injury severities in vehicle-bicycle crashes, Behnood and Mannering [19] applied a random parameter multinomial logit model. The model estimation results identified various factors which could potentially contribute to severe injuries in vehicle-bicycle crashes. In recent years, the non-parametric classification algorithms have been increasingly adopted for crash severity analysis. Li et al. [20] explored the possibility of using support vector machine (SVM) to analyze crash injury severity. By comparing the SVM model and ordered probit (OP) model results, the authors claimed that the SVM model is slightly superior to the OP model. Chang and Wang [21] developed a classification and regression tree (CART) model to explore the relationship between various factors and injury severity. The results indicated that the vehicle type is the single most important variable associated with injury severity. Li et al. [22] also adopted CART to analyze injury severity of bus passengers with different movements.

As indicated in the literature, existing studies have proposed various methodologies to analyze injury severity. Nonetheless, few studies have considered the data imbalance issue embedded in traffic crash data. In a typical traffic crash dataset, the number of severe injury crashes (minority instances) is much smaller than that of minor injury crashes (majority instances). If left untreated, this data imbalance issue would degrade the performance of standard classifiers, such as logistic regression, CART, and SVM. This is mainly because the standard classifiers are suitable for training

balanced data, and the learning process is guided by achieving the highest overall accuracy [23]. This would induce a bias toward the majority instances. The trained model would classify the majority instances much more accurately while misclassifying the minority instances, making the model fail to be informative [24,25]. When the identification of minority instances is of interest, this misclassification could result in substantial costs. To overcome the data imbalance problem, this study adopts a hybrid data resampling approach which combines the over- and under-sampling.

This paper aims to investigate factors affecting crash severity of real-time ridesharing vehicles based on the CART model. The current study contributes to existing literature in two aspects. First, it enriches previous crash severity studies by explicitly analyzing factors affecting real-time ridesharing vehicle crash severity. Secondly, by introducing a hybrid data resampling approach, the current study significantly improves the classification accuracy of minority instances (severe injury crashes). By accurately identifying factors affecting real-time ridesharing vehicle crash severity, corresponding engineering and administrative countermeasures could be implemented to mitigate the crash severity. This could promote the development of real-time ridesharing service, as well as the sustainability of urban transportation. The remainder of this paper is organized as follows: Section 2 introduces the dataset and provides related descriptive statistics. Section 3 describes the proposed methodology details. Section 4 presents the model results discussion. Section 5 concludes the current study, and the study limitations are also pointed out.

2. Data Preparation

For the modeling purpose, the police-reported crash data within the jurisdiction of City of Chicago from January to December 2018 is obtained [26]. As one of the most densely populated cities, Chicago has a booming real-time ridesharing market. All major companies are operating in Chicago, such as Uber, Lyft, and Via. The fast-developing market has also brought some traffic safety concerns. During 2018, a total of 2624 traffic crashes involve real-time ridesharing vehicles, including 2564 minor injury crashes, 58 incapacitating injury crashes, and 2 fatal crashes. Since there are only 2 fatal crashes, fatal crashes and incapacitating injury crashes are combined as severe injury crashes. Moreover, severe injury crashes account for 2.29% of the total ridesharing vehicle crashes. It is worth noting that in the same period, the average percentage of severe injury crashes in Chicago is 1.87%, which further emphasizes the necessity to improve the safety of real-time ridesharing service and make it a more reassuring travel mode.

In the current study, the dependent variable is crash severity, which is a binary variable (minor injury or severe injury). The original dataset contains detailed information on each crash. Based on the scope of this study, 15 independent variables are extracted, which are divided into 5 categories, including vehicle-related variables, driver-related variables, infrastructure-related variables, environment-related variables, and crash attributes. Please refer to Table 1 for the detailed variable description and corresponding distribution.

Table 1. Variable description and corresponding distribution.

Variables	Description of Variables	No. of Crashes	Distribution
Crash severity	Minor injury = 0	2564	97.71%
	Severe injury = 1	60	2.29%
Vehicle-related			
Vehicle manufacture year	2014–2018 = 1	1629	62.08%
	2009–2013 = 2	647	24.66%
	< 2009 = 3	348	13.26%
Vehicle type	Passenger car = 1	2133	81.29%
	Sport utility vehicle = 2	491	19.07%
Vehicle maneuver	Changing lanes = 1	105	4.00%
	Entering traffic from parking = 2	82	3.13%
	Parked = 3	98	3.73%
	Overtaking = 4	76	2.90%
	Slow/stop = 5	443	16.88%
	Straight ahead = 6	1258	47.94%
	Turning = 7	317	12.08%
	Other = 8	245	9.34%
Number of passengers	No passenger = 0	53	1.89%
	One passenger = 1	1283	49.28%
	Two passengers = 2	863	32.54%
	Three or more passengers = 3	425	16.28%
Driver-related			
Gender	Male = 1	2071	78.93%
	Female = 2	553	21.07%
Age	≤ 29 = 1	806	30.72%
	30–45 = 2	962	36.66%
	46–59 = 3	635	24.20%
	≥ 60 = 4	221	8.42%
Driver action	Improper action (improper lane change, failed to yield, etc.) = 1	752	28.66%
	No improper action = 2	1872	71.34%
Infrastructure-related			
Traffic control device	No controls = 1	1247	47.52%
	Stop sign = 2	285	10.86%
	Traffic signal = 3	999	38.07%
	Other = 4	93	3.54%
Trafficway type	Divided with median (not raised) = 1	456	17.38%
	Divided with median barrier = 2	253	9.64%
	Not divided = 3	1366	52.05%
	One-way = 4	504	19.21%
	Parking lot = 5	45	1.71%
Intersection	Intersection = 1	839	31.97%
	Non-intersection = 2	1785	68.03%
Roadway surface condition	Dry = 1	2061	78.54%
	Wet = 2	448	17.07%
	Snow/slush = 3	88	3.35%
	Other = 4	27	1.03%
Environment-related			
Weather condition	Clear = 1	2055	78.32%
	Cloudy = 2	85	3.24%
	Rain = 3	283	10.79%
	Snow = 4	201	7.66%
Lighting condition	Darkness = 1	138	2.02%
	Darkness, lighted road = 2	914	26.78%
	Daylight = 3	1489	66.33%
	Dusk = 4	83	4.86%
Crash attribute			
Time	AM peak (07:00–08:59) = 1	226	8.61%
	PM peak (18:00–19:59) = 2	359	13.68%
	Non-peak = 3	2039	77.71%
Pedestrian/pedalcyclist involvement	No = 0	2501	95.31%
	Yes = 1	123	4.69%

3. Proposed Methodology

The Classification and Regression Tree (CART) method is used to identify factors affecting real-time ridesharing vehicle crash severity. However, the data imbalance issue needs to be properly handled before applying CART. As can be seen from Table 1, the crash severity in the current study is quite imbalanced: only 2.29% crashes are severe injury crashes. Without reasonable treatment, the imbalanced data would seriously undermine the model performance.

3.1. Data Imbalance Treatment

The imbalanced learning has attracted attentions of researchers from various fields in the past decades, such as diseases diagnosis [27,28], chemical engineering [29], and geosciences and remote sensing [30,31]. To cope with this problem, hundreds of algorithms have been proposed, which fall into two categories: data preprocessing and cost-sensitive learning. Data preprocessing is generally conducted to attain more balanced input data before training the classification model. Resampling is the most commonly used data preprocessing technique. On the other hand, the cost-sensitive learning technique aims to modify learning algorithms by assuming higher costs for the misclassification of minority instances regarding majority instances. However, it can be tricky to set values of the misclassification costs. Please refer to [23] for a detailed review of methods used in imbalanced data classification.

The current study adopts SMOTE+ENN, a hybrid data preprocessing approach, to produce better-defined class clusters [32]. The motivation behind this approach is to combine the data over-sampling and under-sampling method by sequentially applying SMOTE (Synthetic Minority Over-sampling Technique) and ENN (Edited Nearest Neighbor) algorithm. Initially proposed by Chawla et al. [33], SMOTE is a widely used over-sampling algorithm. As illustrated in Figure 1, the SMOTE algorithm aims to create a more balanced dataset by randomly generating artificial minority instances on the line segments connecting each minority instance with its k nearest neighbors. This is done in the following way. First, SMOTE calculates the difference between the feature vector (an n-dimensional vector representing variables in the problem) under consideration and its k nearest neighbor (in this case, $k = 3$). Secondly, this difference is multiplied by a random number between 0 and 1, which is then added to the feature vector. This will randomly select a point and create an artificial minority instance along the line segment connecting two feature vectors. Depending on the number of artificial instances required, the value of k can be specified.

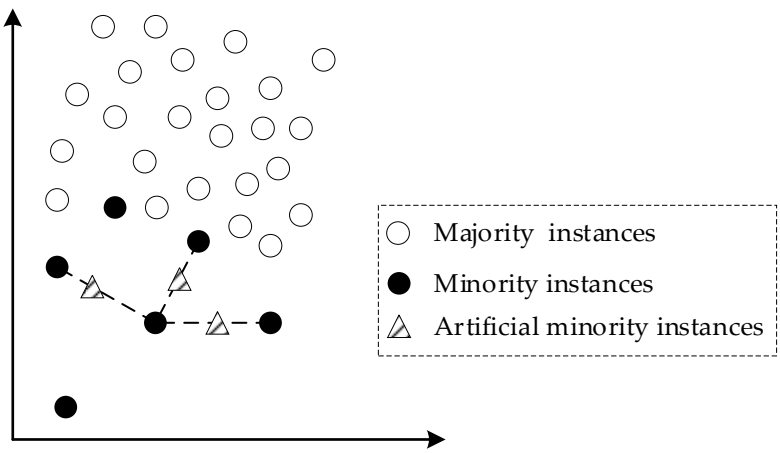

Figure 1. Illustration of Synthetic Minority Over-sampling Technique (SMOTE) algorithm: generating artificial minority instances.

Although the SMOTE algorithm could generate a more balanced class distribution, it can also induce other problems. As can be seen from Figure 2b, the interpolation of minority instances (solid

dots) could create artificial minority instances too deeply in the majority instance cluster (hollow dots). Under such a situation, applying a classification algorithm might result in overfitting. To solve this problem and create better-defined class clusters, the ENN algorithm is applied to the over-sampled dataset for the data cleaning purpose. The ENN algorithm could be described in the following way [34]:

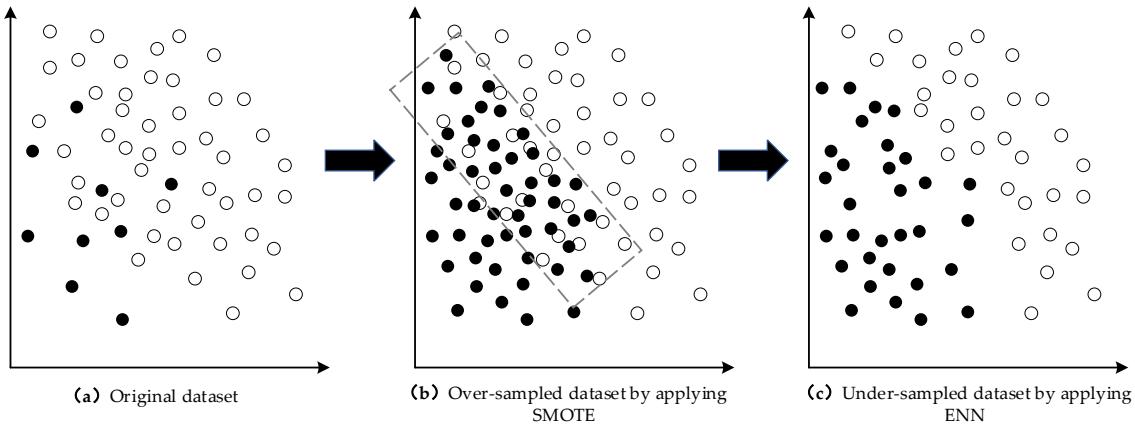

Figure 2. Illustration of SMOTE+ Edited Nearest Neighbor (ENN) method: generating a more balanced class distribution by combining data over-sampling and under-sampling.

For each sample E_i, its three nearest neighbors are identified;

If E_i belongs to the minority class, and the class labels of at least two of E_i's three nearest neighbors are majority, then E_i is removed;

If E_i belongs to the majority class, and the class labels of at least two of E_i's three nearest neighbors are minority, then E_i is removed.

By removing samples from both classes, the ENN algorithm could provide in-depth data cleaning, resulting in better-defined class clusters (Figure 2c).

Please refer to Figure 2 for the illustration of SMOTE+ENN method.

After applying the SMOTE+ENN method to preprocess the original data, the artificially more balanced dataset could be used to train the CART model. In the current paper, the SMOTE+ENN method is performed based on the Python toolbox imbalanced-learn [35].

3.2. Classification and Regression Tree

As one of the most popular data mining algorithms, CART has been widely employed in various classification tasks. By learning the decision rules inferred from a set of sample features, CART can predict the label of the target sample. As a non-parametric supervised learning algorithm, CART has several advantages in crash severity analysis. The primary advantage is that the CART model does not impose any presumed relationship between crash severity and corresponding contribution factors. Besides, the CART model could effectively handle the multicollinearity problem, which is a common issue in traffic crash data. Additionally, the analysis results can be visualized as a tree structure, which will make the results more easily interpreted by transportation officials. For a continuous target variable, a regression tree should be built. And a classification tree can be used for a categorical target variable. In this study, the target variable is binary (minor injury or severe injury), and a classification tree is developed.

The development of a CART model generally involves two steps: tree growing and tree pruning. Tree growing starts at the root node, which includes all the training data. The idea behind tree growing is to partition the target variable recursively to minimize the impurity of the two resultant child nodes. During each step, the CART model aims to identify a splitter (independent variable) which leads to the most significant improvement in the purity of two child nodes. Several splitting criteria are available to measure purity improvement. The Gini index, which is the most common measure of impurity, is

used in the current study. If a node m is partitioned into two child nodes n_1 and n_2 by a splitter θ, the Gini index for any child node can be calculated as:

$$H(n(\theta)) = 1 - \sum_c p\left(c\middle|n\right)^2, n \in (n_1, n_2) \tag{1}$$

where $p(c|n)$ indicates the proportion of class c instances in a child node n. And the impurity of node m could be calculated as:

$$G(\theta) = \frac{t_1}{N_m} H(n_1(\theta)) + \frac{t_2}{N_m} H(n_2(\theta)) \tag{2}$$

where t_1 and t_2 are numbers of instances in child node n_1 and n_2; N_m is the total number of instances in node m. By checking all possible splitters, the CART model selects the splitter which can minimize the value of $G(\theta)$ to generate two child nodes. As shown in Figure 3, all crashes in root node m are first split into child nodes n_1 and n_2 by the best possible splitter—lighting condition. Then, crashes occurred in darkness are further divided into node n_3 and n_4, depending on the gender of drivers. This would create child nodes which are as homogenous as possible.

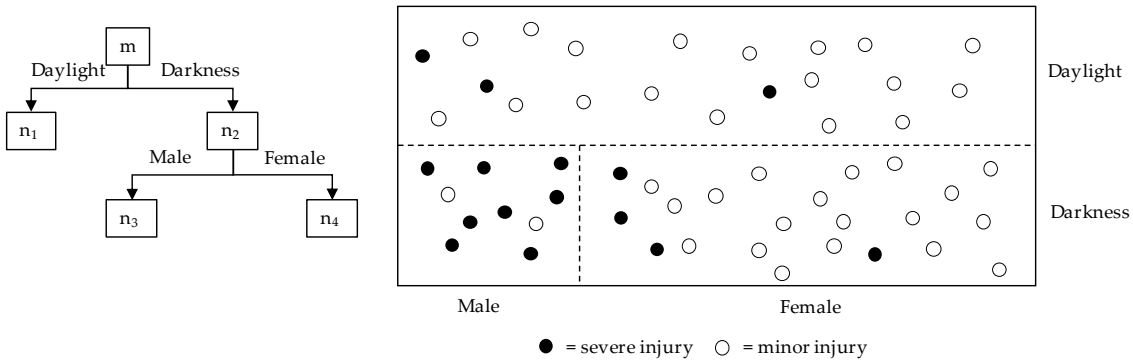

Figure 3. Example of tree growing in classification and regression tree (CART): the process of partitioning the target variable recursively to minimize the impurity of child nodes.

By recursively partitioning the target variable based on the Gini index, the tree keeps growing until it is impossible to divide any node. At which point, all instances within each leaf node are homogenous and a saturated tree is created. However, the saturated tree is most likely overfitting. This means the tree fits the current data too closely and could result in high misclassification rate when applied to a new dataset. Therefore, the saturated tree needs to be pruned. The principle of tree pruning is to cut off branches which contribute little to the classification performance of the tree. In this study, the saturated tree is pruned by tuning parameters which control the tree growing process. The CART model is coded in Python based on the popular machine learning library scikit-learn [36]. During the training of the CART model, the program could tune several parameters in order to avoid the overfitting problem and promote the classification performance. Please refer to Table 2 for the details of parameters to be tuned.

Table 2. Parameters to be tuned.

Parameter	Definition	Value Range
min_sample_split	The minimum number of samples required to split an internal node	Between 2 and 400
max_depth	The maximum depth of a tree	Between 4 and 15
max_leaf_nodes	The maximum number of leaf nodes in a tree	Between 5 and 20
min_samples_leaf	The minimum number of samples required to be at a leaf node	Between 2 and 500

The randomized search parameter tuning approach is utilized to find the optimal parameter set. The program is set to randomly generate 1000 different parameter sets based on the value range of each parameter. Each parameter set can generate a corresponding tree, and the classification performance of which is evaluated by the 5-fold cross-validation. To evaluate a tree, all the original data is first randomly divided into five groups. During each step of the 5-fold cross-validation, one group of data is selected as the testing data, and the remaining four groups of data are combined to train the model. Before training the model, the training data is preprocessed based on the SMOTE+ENN method to generate a more balanced artificial dataset, which is then used to train the model. It is imperative to notice that the artificially more balanced data is only used for model training. The model performance is still evaluated with the imbalanced testing data. This process is repeated five times, and the average classification accuracy is reported. Since the model is tested against an independent dataset during each step, the reported classification accuracy could reflect the ability of the model to classify unseen data. After 1000 iterations, the program will report the optimal model with the highest classification accuracy.

Besides, the CART model trained with the original imbalanced data is used as the baseline model for comparison. For the following part, the baseline model is denoted as $CART_0$, and the model trained on the artificially more balanced data is denoted as $CART_1$.

3.3. Classification Performance Evaluation

To compare the classification performance of the $CART_0$ and $CART_1$ model, three most commonly used evaluation metrics for imbalanced data classification are adopted: Receiver Operating Characteristics (ROC) curve, geometric mean (G-mean), and accuracy [23].

Accuracy is the most intuitionistic and general metric for classification model evaluation. Based on the confusion matrix shown in Table 3, the accuracy can be easily calculated.

Table 3. Confusion matrix.

	Predicted Positive	Predicted Negative
Actual positive	True positive (TP)	False negative (FN)
Actual negative	False positive (FP)	True negative (TN)

In Table 3, TP and TN are the number of positive and negative samples correctly classified by the model. FP and FN are the number of negative and positive samples misclassified by the model. The accuracy is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{3}$$

However, when the data imbalance is present, accuracy might not be a right choice due to the bias toward the majority instances. If the majority instances significantly outnumber minority instances, the model might still report extremely high accuracy even when all minority instances are misclassified.

On the other hand, ROC curve and G-mean are less likely to suffer from this problem as they take class distribution into consideration. ROC curve is a visualization of the trade-off between the true positive rate (TPR) and false positive rate (FPR), which are defined as follows:

$$TPR = \frac{TP}{TP + FN} \tag{4}$$

$$FPR = \frac{FP}{FP + TN} \tag{5}$$

TPR is represented by the X axis of the ROC curve, and FPR is represented by the Y axis. The CART model could report a score which represents the degree to which a randomly picked sample is a member of a class. Each score corresponds to a specific set of TPR and FPR, which is represented by a point in the ROC space. By changing the threshold value of a sample belonging to a class, different sets of TPR and FPR could be generated, resulting in different points in the ROC space. A ROC curve could be produced by connecting these points. The area under the ROC curve (AUC) measures how well a model could distinguish between classes. The AUC score is between 0 and 1, and higher AUC represents better classification performance.

The other metric used to evaluate the imbalanced data classification performance is G-mean, which is calculated as:

$$G - mean = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}} \tag{6}$$

G-mean is not affected by the size of the majority and minority classes. By balancing the accuracy of each class, G-mean attempts to maximize the accuracy of both classes.

4. Results and Discussions

Table 4 summarizes the parameter tuning results for the $CART_0$ and $CART_1$ model.

Table 4. Results of parameter tuning.

	$CART_0$ Model	**$CART_1$ Model**
min_sample_split	21	375
max_depth	5	14
max_leaf_nodes	10	13
min_samples_leaf	3	7
Program running time	10.04 s	711.57 s

After parameter tuning, the optimal $CART_0$ and $CART_1$ model are applied to the original crash data, respectively. To compare the classification performance, the confusion matrixes of both models are reported in Table 5.

Table 5. Confusion matrix of $CART_0$ and $CART_1$ model.

	$CART_0$ Model		**$CART_1$ Model**	
	Predicted Minor Injury	**Predicted Severe Injury**	**Predicted Minor Injury**	**Predicted Severe Injury**
Actual minor injury	2564	0	2065	499
Actual severe injury	60	0	20	40

According to Equations (3) and (6), the accuracy and G-mean for both models could be calculated. The accuracies of $CART_0$ and $CART_1$ model are 98% and 82%, respectively. By classifying all crashes as minor injury crashes, the accuracy of the $CART_0$ model is exceptionally high. Nevertheless, no severe injury crash is correctly predicted, which severely undermines the model's practicality. Although the accuracy of the $CART_1$ model is 82%, which is 16% lower than that of the $CART_0$ model, it successfully

predicts 40 out of 60 severe injury crashes. On the other hand, the G-mean of $CART_0$ and $CART_1$ model are 0% and 73%, respectively, demonstrating a far more superior classification performance of the $CART_1$ model. Again, this is because the $CART_0$ model misclassifies all the severe injury crashes.

Besides, the ROC curve and AUC of both models are also reported by the program, which is shown in Figure 4.

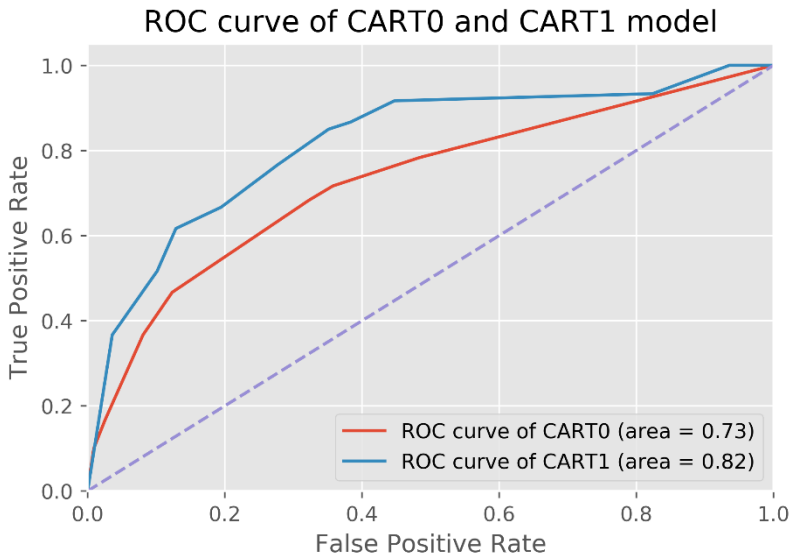

Figure 4. Receiver Operating Characteristics (ROC) curve of $CART_0$ and $CART_1$ model.

The ROC curve further reveals that $CART_1$ model outperforms $CART_0$ model. By preprocessing the crash data with SMOTE+ENN, the AUC is improved from 0.73 to 0.82, a 12% increase. The classification tree generated by $CART_1$ model is shown in Figure 5.

The interpretation of the classification tree is straightforward. The tree growing starts at node 0, containing all the data which has been preprocessed by the SMOTE+ENN method. As can be seen from node 0, the minor injury (MI) and severe injury (SI) crashes are much more balanced. The tree has thirteen terminal nodes, which are marked as grey boxes in Figure 5. It can be easily distinguished that pedestrian/pedalcyclist involvement, number of passengers, weather condition, trafficway type, vehicle manufacture year, traffic control device, the gender of driver, lighting condition, vehicle type, driver age and crash time are the primary factors affecting crash severity of real-time ridesharing vehicles. The initial split at node 0 is based on pedestrian/pedalcyclist involvement. The tree directs crashes involving pedestrian/pedalcyclist to the right, forming node 2, and other crashes are directed to the left, forming node 1. Node 2 is further split into terminal node 5 and 6 according to the weather condition. The tree sends crashes occurred on snowy days to terminal node 6 and crashes occurred on other weather conditions (clear, cloudy or rain) to terminal node 5. As indicated by terminal node 5 and 6, crashes involving pedestrians/pedalcyclist are highly likely to be severe injury crashes in almost all weather conditions except for snow (99% versus 0%). This is probably because the icy pavement caused by snowfall would force the drivers to slow down. As such, severe injury crashes might be avoided. On the left branch of the tree, node 1 is further divided into node 3 and node 4 based on the number of passengers. Ridesharing vehicles with two or more passengers are more prone to severe injury crashes compared with vehicles with one or no passenger (59% versus 14%). With more passengers in the car, the driver might be distracted by chats among passengers, which could increase the possibility of severe injury crashes. The tree splits nodes 3 into node 7 and terminal node 8 based on the trafficway type. Compared with other trafficway types, crashes occurred on a divided road with median are more likely to be severe injury crashes (39% versus 4%). Divided roads with median treatments are generally associated with higher traffic volume and faster travel speed, which might result in more severe crashes. Node 4 is further split into node 9 and 10 according to the vehicle

manufacture year. Compared with vehicles manufactured prior to 2013, vehicles produced between 2014 and 2018 show a greater propensity toward severe injury crashes (70% versus 24%). Drivers with relatively new vehicles probably purchase these vehicles recently to provide real-time ridesharing service, which means they are lack of driving experience. As such, these drivers are more likely to be involved in severe injury crashes. Based on traffic control devices, node 7 is split into terminal node 11 and 12. Compared with crashes occurred at places with some traffic control device (stop sign/traffic signal/other), crashes occurred at places without any traffic control are highly unlikely to be severe injury crashes (0% versus 57%). This could also be due to travel speeds at places without any traffic control are relatively slow, resulting in less severe crashes. The tree continues to divide node 9 into node 13 and terminal node 14 based on the gender of drivers. As expected, male drivers are much likely to be involved in severe injury crashes compared with female drives (75% versus 10%). Node 10 is split into terminal node 15 and 16 according to the lighting condition. Compared with daylight or dusk, darkness could significantly increase the risk of severe injury crashes (46% versus 2%). Node 13 is further divided into node 17 and terminal node 18 based on the type of vehicles providing real-time ridesharing service. Compared with passenger cars, sport utility vehicles (SUV) are less prone to severe injury crashes (24% versus 78%). The is probably because SUV has a high center of gravity and is usually reinforced to protect the occupants. The tree continues to split node 17 into node 19 and 20 based on the lighting condition. Crashes occurred in darkness are directed to the left, forming node 19. And crashes occurred in daylight or at dusk are directed to the right, forming node 20. Based on the driver age, node 19 is further divided into terminal node 21 and 22. Conditioned on previous splitters, drivers older than 60 are highly unlikely to be involved in severe injury crashes compared with younger drivers (0% versus 88%). This may be due to older drivers have more driving experiences and tend to take fewer risks while driving. Node 20 is split into terminal node 23 and 24 according to the crash time. Compared with non-peak hours, peak hours are more prone to severe injury crashes (84% versus 48%). During peak hours, there is greater demand for real-time ridesharing service. Drivers tend to drive faster and even recklessly to serve as many customers as possible, which would increase the risk of more severe crashes.

Out of the thirteen terminal nodes, four of them (node 5, 12, 21 and 23) predict that the crash is more likely to be severe injury crashes, namely, the proportion of severe injury crashes is higher than that of minor injury crashes. The associated decision rules are extracted as follows:

Involving pedestrians/pedalcyclists & clear/cloudy/rain weather;

Not involving pedestrians/pedalcyclists & number of passengers ≤1 & divided road with median treatment & stop sign/traffic signal/other traffic control device;

Not involving pedestrians/pedalcyclists & number of passengers ≥2 & vehicle manufactured during 2014–2018 & male driver & passenger car & darkness & driver age ≤59;

Not involving pedestrians/pedalcyclists & number of passengers ≥2 & vehicle manufactured during 2014–2018 & male driver & passenger car & daylight/dusk & peak hour.

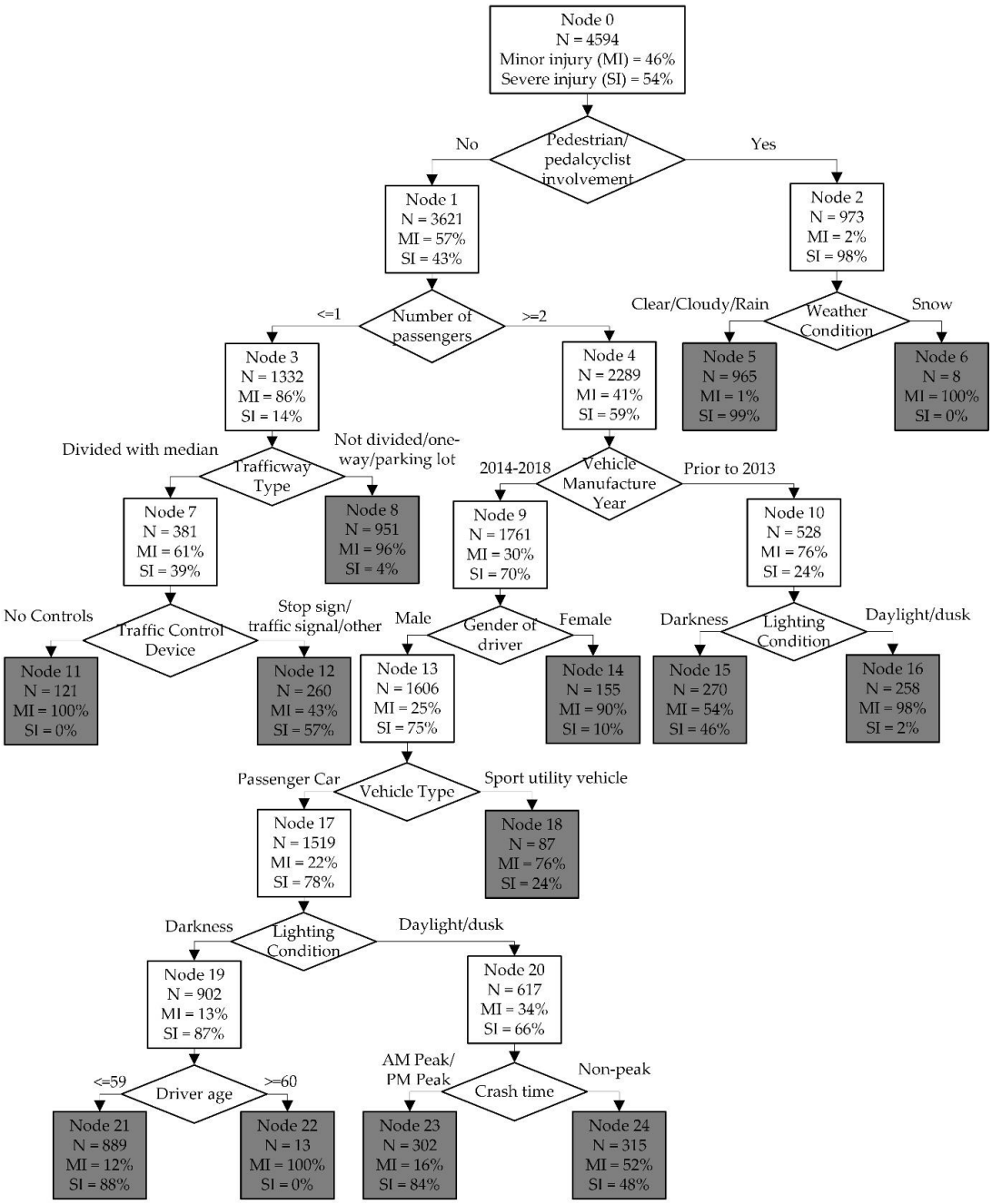

Figure 5. Classification tree generated by $CART_1$ model.

5. Conclusions

As a fast-growing travel mode, real-time ridesharing service has promoted the social and environmental sustainability in various ways. However, the booming market has also brought some traffic safety concerns, which haven't been thoroughly investigated in previous literature. Without accurately identifying factors affecting crash severity of real-time ridesharing vehicles, the development of this travel mode and urban transportation sustainability might be affected. Aiming to explore factors affecting real-time ridesharing vehicle crash severity, this paper has utilized the CART model to extract associated decision rules. The Chicago police-reported crash data from January to December 2018 is collected. Crash severity in the original dataset is highly imbalanced. Over 97% of crashes are minor injuries, and severe injuries account for only 2.29% of total crashes. This would induce a bias toward the minor injuries and undermine the performance of the CART model. To solve the data

imbalance problem and promote the classification performance of the CART model, SMOTE+ENN is introduced to preprocess the original data. By sequentially applying SMOTE and ENN algorithm, this hybrid approach can produce more balanced class clusters. The artificially more balanced dataset is then used to train the CART model. It should be noticed that the artificial data is only used for model training, and the performance of the model is still tested with the imbalanced data. The CART model trained with original imbalanced data is selected as the baseline model for comparison. After parameter tuning, both models are applied to classify the original data.

By preprocessing the crash data with SMOTE+ENN, the classification accuracy is reduced from 98% to 82%. However, the baseline model misclassifies all severe injury crashes, and the proposed methodology could successfully predict 40 out of 60 severe injury crashes. Additionally, the G-mean is increased from 0% to 73%, and the AUC is increased from 0.73 to 0.82, which demonstrates that the proposed methodology has a better classification performance than the baseline model. The tree generated by the proposed methodology reveals that following variables are the primary indicators of real-time ridesharing vehicle crash severity: pedestrian/pedalcyclist involvement, number of passengers, weather condition, trafficway type, vehicle manufacture year, traffic control device, gender of driver, lighting condition, vehicle type, driver age and crash time. Out of the thirteen terminal nodes, four of them (node 5, 12, 21 and 23) indicate that the crash is more likely to be severe injury crashes. Moreover, the associated decision rules are extracted.

The results of the current study could provide some valuable insights for the development of safety improvement programs focusing on real-time ridesharing vehicles. For instance, when two or more passengers are present, the ridesharing app should remind the driver not to be distracted by passengers' chatting. Moreover, when a driver with a relatively new vehicle registers to provide real-time ridesharing service, the company should provide reinforced safety education to him/her. These potential countermeasures could mitigate the crash severity and help to fully utilize the benefits of real-time ridesharing on urban transportation sustainability.

Although the proposed methodology presents superior classification performance, the methodology limitations should be noted. For instance, the CART model cannot provide the confidence interval of risk factors. Besides, it is usually challenging to conduct a sensitivity analysis with the CART model. The injury severity is a binary variable in the current study. When the injury severity is classified into three or more categories, the performance of the proposed methodology needs to be reevaluated. Since the current paper only uses one-year crash data, future studies could benefit from collecting more abundant crash data with more explanatory variables. The SMOTE+ENN method is independent of classification models, so future studies should consider combining SMOTE+ENN with other classifiers to further evaluate the data preprocessing effect.

Author Contributions: Conceptualization, B.Z. and X.Z.; Data curation, B.Z. and X.Z.; Funding acquisition, B.Z., S.Z. and Z.L.; Methodology, B.Z., X.Z. and S.Z.; Software, B.Z.; Validation, B.Z., X.Z. and X.L.; Visualization, X.L.; Writing—original draft, B.Z.; Writing—review & editing, Z.L.

Funding: This research was funded by China Postdoctoral Science Foundation, grant number 2015M582593, the Natural Science Basic Research Plan in Shaanxi Province of China, grant number 2018JQ5147, National Natural Science Foundation of China, grant number 71871029, and the Fundamental Research Funds for the Central Universities, CHD, grant number 300102218401, 300102219306, 300102218404.

Acknowledgments: We would like to express our gratitude to the Chicago Police Department for making the data used in this study publicly available.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Amey, A.; Attanucci, J.; Mishalani, R. Real-Time Ridesharing: Opportunities and Challenges in Using Mobile Phone Technology to Improve Rideshare Services. *Transp. Res. Rec. J. Transp. Res. Board* **2011**, *2217*, 103–110. [CrossRef]

2. Barrios, J.M.; Hochberg, Y.V.; Yi, H. The Cost of Convenience: Ridesharing and Traffic Fatalities. 2018. Available online: http://dx.doi.org/10.2139/ssrn.3259965 (accessed on 19 April 2019).
3. Furuhata, M.; Dessouky, M.; Ordóñez, F.; Brunet, M.E.; Wang, X.; Koenig, S. Ridesharing: The state-of-the-art and future directions. *Transp. Res. Part B Methodol.* **2013**, *57*, 28–46. [CrossRef]
4. Yu, B.; Ma, Y.; Xue, M.; Tang, B.; Wang, B.; Yan, J.; Wei, Y.M. Environmental benefits from ridesharing: A case of Beijing. *Appl. Energy* **2017**, *191*, 141–152. [CrossRef]
5. Ma, R.; Zhang, H.M. The morning commute problem with ridesharing and dynamic parking charges. *Transp. Res. Part B Methodol.* **2017**, *106*, 345–374. [CrossRef]
6. Amirkiaee, S.Y.; Evangelopoulos, N. Why do people rideshare? An experimental study. *Transp. Res. Part F Traffic Psychol. Behav.* **2018**, *55*, 9–24. [CrossRef]
7. Nourinejad, M.; Roorda, M.J. Agent based model for dynamic ridesharing. *Transp. Res. Part C Emerg. Technol.* **2016**, *64*, 117–132. [CrossRef]
8. Agatz, N.; Erera, A.; Savelsbergh, M.; Wang, X. Optimization for dynamic ride-sharing: A review. *Eur. J. Oper. Res.* **2012**, *223*, 295–303. [CrossRef]
9. Schreieck, M.; Safetli, H.; Siddiqui, S.A.; Pflügler, C.; Wiesche, M.; Krcmar, H. A Matching Algorithm for Dynamic Ridesharing. *Transp. Res. Procedia* **2016**, *19*, 272–285. [CrossRef]
10. Simonetto, A.; Monteil, J.; Gambella, C. Real-time city-scale ridesharing via linear assignment problems. *Transp. Res. Part C Emerg. Technol.* **2019**, *101*, 208–232. [CrossRef]
11. Alonso-mora, J.; Samaranayake, S.; Wallar, A.; Frazzoli, E.; Rus, D. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proc. Natl. Acad. Sci. USA* **2017**, *114*, 462–467. [CrossRef]
12. Savolainen, P.T.; Mannering, F.L.; Lord, D.; Quddus, M.A. The statistical analysis of highway crash-injury severities: A review and assessment of methodological alternatives. *Accid. Anal. Prev.* **2011**, *43*, 1666–1676. [CrossRef] [PubMed]
13. Anastasopoulos, P.C.; Shankar, V.N.; Haddock, J.E.; Mannering, F.L. A multivariate tobit analysis of highway accident-injury-severity rates. *Accid. Anal. Prev.* **2012**, *45*, 110–119. [CrossRef] [PubMed]
14. Yu, R.; Abdel-Aty, M. Analyzing crash injury severity for a mountainous freeway incorporating real-time traffic and weather data. *Saf. Sci.* **2014**, *63*, 50–56. [CrossRef]
15. Chen, C.; Zhang, G.; Tarefder, R.; Ma, J.; Wei, H.; Guan, H. A multinomial logit model-Bayesian network hybrid approach for driver injury severity analyses in rear-end crashes. *Accid. Anal. Prev.* **2015**, *80*, 76–88. [CrossRef] [PubMed]
16. Haleem, K.; Alluri, P.; Gan, A. Analyzing pedestrian crash injury severity at signalized and non-signalized locations. *Accid. Anal. Prev.* **2015**, *81*, 14–23. [CrossRef] [PubMed]
17. Naik, B.; Tung, L.W.; Zhao, S.; Khattak, A.J. Weather impacts on single-vehicle truck crash injury severity. *J. Saf. Res.* **2016**, *58*, 57–65. [CrossRef] [PubMed]
18. Zeng, Z.; Zhu, W.; Ke, R.; Ash, J.; Wang, Y.; Xu, J.; Xu, X. A generalized nonlinear model-based mixed multinomial logit approach for crash data analysis. *Accid. Anal. Prev.* **2017**, *99*, 51–65. [CrossRef]
19. Behnood, A.; Mannering, F. Determinants of bicyclist injury severities in bicycle-vehicle crashes: A random parameters approach with heterogeneity in means and variances. *Anal. Methods Accid. Res.* **2017**, *16*, 35–47. [CrossRef]
20. Li, Z.; Liu, P.; Wang, W.; Xu, C. Using support vector machine models for crash injury severity analysis. *Accid. Anal. Prev.* **2012**, *45*, 478–486. [CrossRef]
21. Chang, L.Y.; Wang, H.W. Analysis of traffic injury severity: An application of non-parametric classification tree techniques. *Accid. Anal. Prev.* **2006**, *38*, 1019–1027. [CrossRef]
22. Li, D.; Zhao, Y.; Bai, Q.; Zhou, B.; Ling, H. Analyzing injury severity of bus passengers with different movements. *Traffic Inj. Prev.* **2017**, *18*, 528–532. [CrossRef]
23. Haixiang, G.; Yijing, L.; Shang, J.; Mingyun, G.; Yuanyue, H.; Bing, G. Learning from class-imbalanced data: Review of methods and applications. *Expert Syst. Appl.* **2017**, *73*, 220–239. [CrossRef]
24. Jeong, H.; Jang, Y.; Bowman, P.J.; Masoud, N. Classification of motor vehicle crash injury severity: A hybrid approach for imbalanced data. *Accid. Anal. Prev.* **2018**, *120*, 250–261. [CrossRef] [PubMed]
25. Zhou, B.; Li, Z.; Zhang, S.; Zhang, X.; Liu, X.; Ma, Q. Analysis of Factors Affecting Hit-and-Run and Non-Hit-and-Run in Vehicle-Bicycle Crashes: A Non-Parametric Approach Incorporating Data Imbalance Treatment. *Sustainability* **2019**, *11*, 1327. [CrossRef]

26. Traffic Crashes—City of Chicago. Available online: https://data.cityofchicago.org/Transportation/Traffic-Crashes-Crashes/85ca-t3if (accessed on 10 March 2019).
27. Dubey, R.; Zhou, J.; Wang, Y.; Thompson, P.M.; Ye, J. Alzheimer's Disease Neuroimaging Initiative Analysis of sampling techniques for imbalanced data: An n=648 ADNI study. *Neuroimage* **2014**, *87*, 220–241. [CrossRef] [PubMed]
28. Bae, S.-H.; Yoon, K.-J. Polyp Detection via Imbalanced Learning and Discriminative Feature Learning. *IEEE Trans. Med. Imaging* **2015**, *34*, 2379–2393. [CrossRef]
29. Raposo, L.M.; Arruda, M.B.; de Brindeiro, R.M.; Nobre, F.F. Lopinavir Resistance Classification with Imbalanced Data Using Probabilistic Neural Networks. *J. Med. Syst.* **2016**, *40*, 69. [CrossRef]
30. Fang, H.; Liang, S. Retrieving leaf area index with a neural network method: Simulation and validation. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 2052–2062. [CrossRef]
31. Svendsen, D.H.; Martino, L.; Campos-Taberner, M.; Garcia-Haro, F.J.; Camps-Valls, G. Joint Gaussian Processes for Biophysical Parameter Retrieval. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 1718–1727. [CrossRef]
32. Batista, G.E.A.P.A.; Prati, R.C.; Monard, M.C. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 20. [CrossRef]
33. Chawla, N.; Bowyer, K.; Hall, L.; Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-sampling Technique Nitesh. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
34. Wilson, D.L. Asymptotic Properties of Nearest Neighbor Rules Using Edited Data. *IEEE Trans. Syst. Man Cybern.* **1972**, *2*, 408–421. [CrossRef]
35. Lemaître, G.; Nogueira, F.; Aridas, C.K. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *J. Mach. Learn. Res.* **2017**, *18*, 1–5.
36. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830. [CrossRef]