Accident Prediction using Big Data Analysis using Ensemble Learning

JUI MHATRE, Kennesaw State University, USA
NIKHITHA POLAVARAPU, Kennesaw State University, USA
SAI HARSHITHA ACHUTA, Kennesaw State University, USA
CHANDANA SARIBALA, Kennesaw State University, USA
DHANUSH NAGANDLA, Kennesaw State University, USA

Many country wide internal roads and national highways have dim lights or no street lights all over the world. We usually observe some turns on roads more prone to accidents than other places. In our paper we used logistic regression, K-nearest neighbours and decision trees that together build an ensemble learning model for predicting these accident zones. For validating the results, five evaluation metrics such as Accuracy, Precision, f-measures, Re-call and Area under curve are used. State of art model for US accident dataset gives F1 score of 57%. We performed severity of accident prediction using ensemble learning and obtained F1 score of 91% and accuracy of 87%.

Additional Key Words and Phrases: accident dataset, big logistic regression,decision tree, KNN, data-preprocessing ,missing features,normalization

ACM Reference Format:
JUI MHATRE, NIKHITHA POLAVARAPU, SAI HARSHITHA ACHUTA, CHANDANA SARIBALA, and DHANUSH NAGANDLA. 2022. Accident Prediction using Big Data Analysis using Ensemble Learning . 1, 1 (May 2022), 6 pages. https://doi.org/XXXXXXX.XXXXXXX

1 RESEARCH STATEMENT AND CONJECTURE

Our research pertains predicting severity of accidents which could occur in the united states. This research uses different machine learning models that helps in predicting the severity of accidents. We use logistic regression, K nearest neighbors and decision trees. Results from these models are combined using min-max technique to form Ensemble learning technique for better prediction results. We used the United States traffic accident dataset. This dataset is preprocessed using data mining technologies and used different classifiers to model training. The dataset after preprocessing will provide more accurate classification results. We used SMOTE technique for solving the class imbalance problem. Four different evaluation metrics are calculated for validating the results.

2 PRELIMINARY LITERATURE SURVEY

The costs of fatalities and injuries due to traffic accidents have a great impact on society. In recent years, road traffic accidents, especially severe vehicle crashes have increased because of the rapid growth of road traffic. Indeed, in recent years much attention has been paid to determine factors that significantly affect the severity of traffic accidents and several approaches have been used to study this problem [15]. The factors that are related to traffic accidents, include; environmental (i.e. weather conditions and road signs), vehicle type its safety, and the characteristics of traffic users. Identifying factors affecting road accidents help create informed decisions for making safer roads. Some factors may be difficult to quantify or collect large scale data on such as driver attentiveness or emotional states. This can lead to unobserved heterogeneity that requires a more sociological approach to study. Other factors such as traffic speed, time of day, or weather conditions are easier to identify and analyze. Examining a wide range of potential factors surrounding accidents may lead to findings on which specific variables contribute to traffic crashes and need more focus in future studies.

The predictive traffic accidents models are considered to be vital in making smart decisions that can lead to avoid accidents on freeways. With the advancements of information technology, machine learning becomes increasingly mature, and useful information without preconditions can be found in databases. Several studies, such as Krishnaveni and Hemalatha [12] Beshah and Hill [13] Chen et al. [14], have investigated machine learning algorithms in transportation-related applications of the causes of accidents. Krishnaveni and Hemalatha [12] conducted a prospective analysis of 34,575 traffic accident events in Hong Kong. In their study, Naive Bayes, AdaBoostM1, J48, PART, and Random Forest classifiers were employed to predict and detect the severity of injury and causes of accidents using WEKA tool. According to the comparison results, the Random Forest classifier outperformed all other algorithms. There are no percentage results. Beshah and Hill [13], employed Naive Bayes, and K-Nearest Neighbors classifiers to build prediction models to assess the injury severity that were used to analyze and predict the role of road-related factors for traffic accident severity. Moreover, they utilized the PART algorithm to present the knowledge in the form of rules, using the WEKA tool [13]. Chen et al. [14] used the SVM models to investigate driver injury severity patterns in rollover crashes using two-year crash data collected in New Mexico. The results showed that the support vector machine (SVM) models produce reasonable predictions and the polynomial kernel outperforms the Gaussian RBF kernel.

Dong et al. [21] used two modules, an unsupervised feature and a supervised fine-tuning module to perform traffic crash prediction. The results showed that the feature learning section classifies interactive information between the explanatory variables and the feature representations, which decreases the dimensionality of the input and preserves the original information. The research study conducted by Najada et al., [17] used Hong Kong's transportation dataset to predict the actual causes of the accidents. They implemented and compared the performance of several classification algorithms in WEKA. Their experimental results showed that Random

Authors' addresses: JUI MHATRE, Kennesaw State University, USA, jmhatre1@students.kennesaw.edu; NIKHITHA POLAVARAPU, Kennesaw State University, USA, npolavar@students.kennesaw.edu; SAI HARSHITHA ACHUTA, Kennesaw State University, USA, sachuta@students.kennesaw.edu; CHANDANA SARIBALA, Kennesaw State University, USA, csaribal@students.kennesaw.edu; DHANUSH NAGANDLA, Kennesaw State University, USA.

Forest surpassed the Naive Bayes and PART algorithm. Similarly, in another study, Chong et al., [15] built the model using ML algorithms to classify the accident injury severity into five categories. They used Artificial Neural Network (ANN), Support Vector Machine (SVM), and Decision Tree to develop the model. The results of their study highlighted that speeding over the lawful limit is the main reason for fatal injury. Moreover, another interesting research study carried out by the authors [18], helped the community in understanding how accident prediction is done using big data mining and data analysis. In addition, the authors also highlighted how to use data sampling to reconstruct the dataset followed by using prepossessing techniques to make data complete and reliable.

The research study conducted by Elfar et al., [19], used neural networks to predict traffic accidents and congestion. Furthermore, they also proposed two predictive models for training the data. The proposed models achieved more accuracy over the other three classification models used in their study. The authors' contributions further guided how their proposed models can be used in various vehicular applications to improve road safety by warning drivers of upcoming traffic slowdowns. Iranitalab [20], experimented that ML algorithms such as linear regression, and Naive Bayes are effectively used to analyze huge datasets to predict road accidents.

By reviewing the literature thoroughly, we highlight manifold research gaps in the existing crash severity prediction systems: i) most of the previous research studies predicted the accident associated with one or two factors only which are not sufficient for a real situation [11]; ii) a significant number of studies does not deal with the class imbalance problem; iii) unobserved heterogeneity; iv) most of the studies solely rely on a single accuracy measure to evaluate the performance of the algorithm. This research aims to fill all the aforementioned research gaps.

Therefore, there is a need to perform a comprehensive analysis that aims to understand the relationship between the influence factors and traffic crash outcomes. In this study, we utilized the full traffic dataset and used big data technology and tools to gain better insight and achieve more accurate results. So, this study focuses on using machine learning techniques to predict accidents.

3 METHODOLOGY

3.1 Data Source Description

The data used in this study is a country wide accident data set that covers 49 states of the United States. It consists of 47 features and approximately 2.8 million accident records were reported. From February 2016 to December 2021, data has been regularly collected utilizing a variety of data providers, including various API's that give streaming traffic event data. These API's transmit traffic events gathered by a range of institutions inside the road-networks, including the US and state departments of transportation, law enforcement agencies, traffic cameras, and traffic sensors.

3.2 Data Preprocessing

Data preprocessing being a major step, we observed a few problems with data which needs to be handled. For some records, start times occur before end times. Moreover, the format for them is yyyy-mm-dd hh:mm:ss, which should be processed into separate columns.

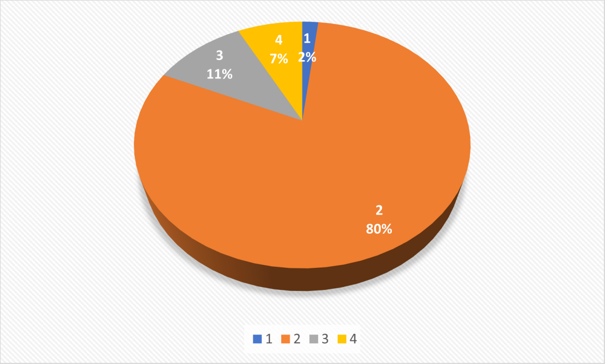


Fig. 1. Imbalance of data for different severity levels

From figure 1 we observe that there is a high imbalance in data where severity level 2 has the highest number of records in the dataset, which account for about 80% followed by 3, 4 and 1. This leads to a need for sampling of data. The range of data varies very largely, so we have normalized the data. Figure 2 gives an overview of missing values for various features. The count of missing values for various columns being very high and predicting or averaging out the missing values would lead to large amount of error. Hence

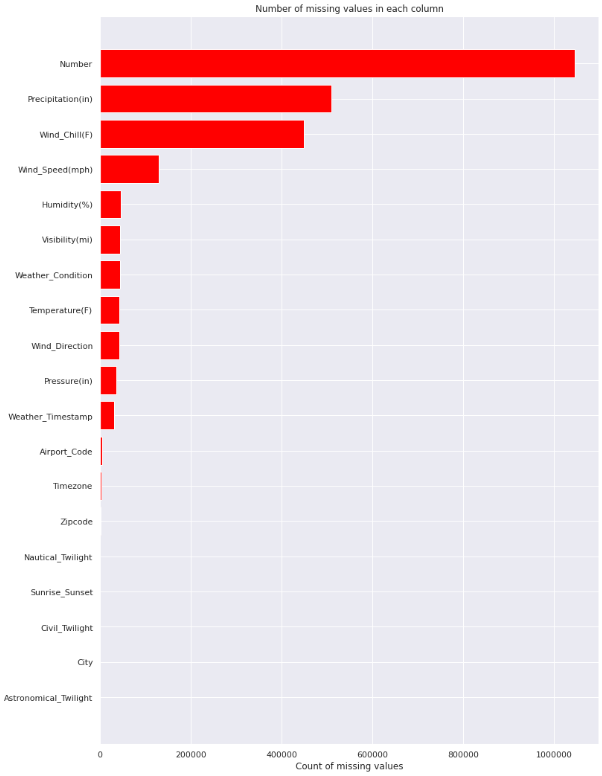


Fig. 2. Figure showing features which have missing values with corresponding missing count

we decided to drop the columns which have missing count more than 5%. Table 1 gives percentages of missing counts for features. We dropped last 4 features due to high missing count.

Features	missing_count	missing (%)
Astronomical_Twilight	83	0.003418623
City	83	0.003418623
Civil_Twilight	83	0.003418623
Sunrise_Sunset	83	0.003418623
Nautical_Twilight	83	0.003418623
Zipcode	935	0.038510996
Timezone	2302	0.094815308
Airport_Code	4248	0.174967605
Weather_Timestamp	30264	1.246520624
Pressure(in)	36274	1.494061893
Wind_Direction	41858	1.724056975
Temperature(F)	43033	1.772453146
Weather_Condition	44007	1.812570483
Visibility(mi)	44211	1.820972882
Humidity(%)	45509	1.874435206
Wind_Speed(mph)	128862	5.307597828
Wind_Chill(F)	449316	18.50653122
Precipitation(in)	510549	21.02861017
Number	1046095	43.08680255

Table 1. Features with missing percentages

3.2.1 Missing features. We handle each missing feature independently. For features like, City, Zip-code, Timezone and Airport code, we have done k-means clustering with cluster number equals count of cities. This forms a cluster of data points corresponding to cities. Missing values are assigned with values equal to the maximum count of values in the corresponding cluster. Euclidean distance between latitude and longitude of given data point is used as a parameter to estimate distance for clustering. Missing values in other features get most frequently occurring values.

3.2.2 Imbalanced dataset. A very high imbalance is observed as we can see in figure 1. This is handled by using Synthetic Minority Oversampling Technique (SMOTE) [8]. It works by randomly picking a point from the minority class and computing the k-nearest neighbors for this point. The synthetic points are added between the chosen point and its neighbors.

3.2.3 Normalization. Scaling of data is very important since it can lead to skewing of our predictions. So, we used standard scaling to scale the feature values.

3.3 Prediction Model

We used three prediction models, Logistic Regression, KNN and Decision Tree combined results in the form of ensemble learning to provide prediction results. The input is feature vectors (Start_lat, Start_lng, Distance (mi), Month, Day, Hour, Weekday, Pressure (in), Temperature (F), Humidity ((mi), Traffic_signal, Junction, Crossing, Stop). The output is traffic accident severity level ranging from 1(low) to 4(high).

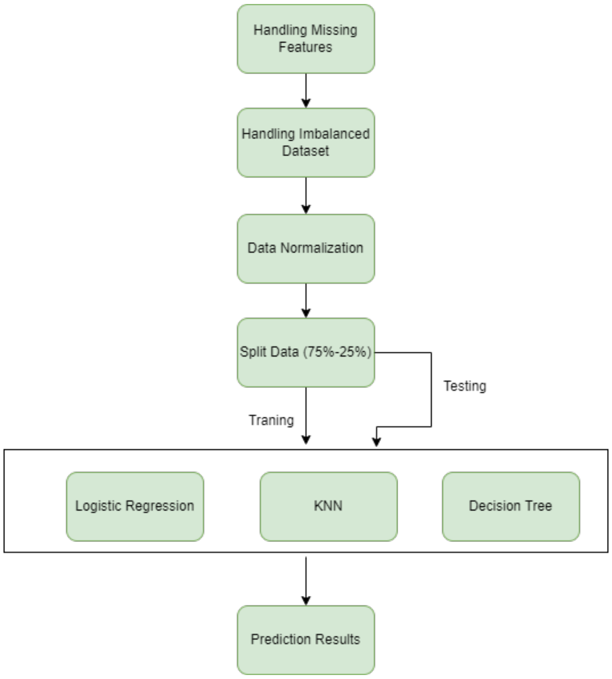
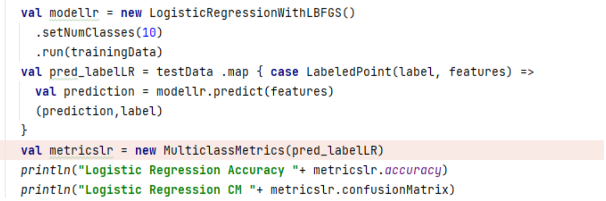


Fig. 3. Methodology

3.3.1 Logistic regression. Limited-memory Broyden–Fletcher –Goldfarb–Shanno (LBFGS) solver only calculates an approximation to the Hessian based on the gradient which makes it computationally more effective. On the other hand it's memory usage is limited compared to regular bfgs which causes it to discard earlier gradients and accumulate only fresh gradients as allowed by the memory restriction. We use L2 penalty for regularization and is supported by lbfgs solver.

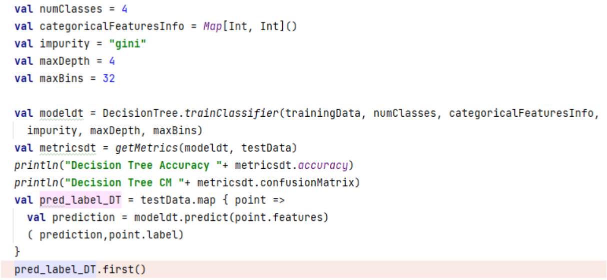


```
val modellr = new LogisticRegressionWithLBFGS()
  .setNumClasses(10)
  .run(trainingData)
val pred_labelLR = testData .map { case LabeledPoint(label, features) =>
  val prediction = modellr.predict(features)
  (prediction,label)
}
val metricslr = new MulticlassMetrics(pred_labelLR)
println("Logistic Regression Accuracy "+ metricslr.accuracy)
println("Logistic Regression CM "+ metricslr.confusionMatrix)
```

Fig. 4. Logistic Regression Code Snippet

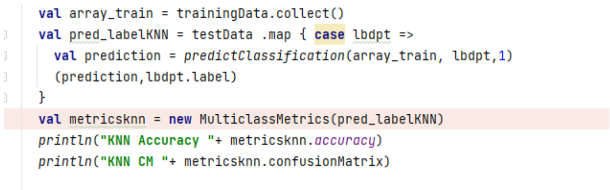
3.3.2 Decision Tree. We have 4 classes which are our severity levels. We use maximum depth of decision trees as 4 and maximum bin size as 32. Impurity parameter used is gini index.

3.3.3 K nearest neighbours. We use this highest neighbour to decide the severity level which requires the K value to be 1. Clusters are created by calculating the euclidean distance among all selected features. Moreover, for convergence, 1000 iterations were required. In Decision Tree we plan to test out model with gini index and entropy with varying hyper-parameters such as min_samples_split,



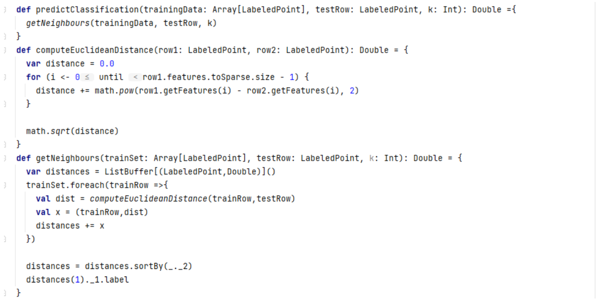
```
val numClasses = 4
val categoricalFeaturesInfo = Map[Int, Int]()
val impurity = "gini"
val maxDepth = 4
val maxBins = 32

val modeldt = DecisionTree.trainClassifier(trainingData, numClasses, categoricalFeaturesInfo,
  impurity, maxDepth, maxBins)
val metricsdt = getMetrics(modeldt, testData)
println("Decision Tree Accuracy "+ metricsdt.accuracy)
println("Decision Tree CM "+ metricsdt.confusionMatrix)
val pred_label_DT = testData.map { point =>
  val prediction = modeldt.predict(point.features)
  ( prediction,point.label)
}
pred_label_DT.first()
```

Fig. 5. Decision Trees Code Snippet



```
val array_train = trainingData.collect()
val pred_labelKNN = testData .map { case lbdpt =>
  val prediction = predictClassification(array_train, lbdpt,1)
  (prediction,lbdpt.label)
}
val metricsknn = new MulticlassMetrics(pred_labelKNN)
println("KNN Accuracy "+ metricsknn.accuracy)
println("KNN CM "+ metricsknn.confusionMatrix)
```

Fig. 6. KNN Code Snippet



```
def predictClassification(trainingData: Array[LabeledPoint], testRow: LabeledPoint, k: Int): Double ={
  getNeighbours(trainingData, testRow, k)
}
def computeEuclideanDistance(row1: LabeledPoint, row2: LabeledPoint): Double = {
  var distance = 0.0
  for (i <- 0 ≤ until < row1.features.toSparse.size - 1) {
    distance += math.pow(row1.getFeatures(i) - row2.getFeatures(i), 2)
  }

  math.sqrt(distance)
}
def getNeighbours(trainSet: Array[LabeledPoint], testRow: LabeledPoint, k: Int): Double = {
  var distances = ListBuffer[(LabeledPoint,Double)]()
  trainSet.foreach(trainRow =>{
    val dist = computeEuclideanDistance(trainRow,testRow)
    val x = (trainRow,dist)
    distances += x
  })

  distances = distances.sortBy(_._2)
  distances(1)._1.label
}
```

Fig. 7. Distance calculation Code Snippet

min_samples_leaf, min_weight_fraction_leaf, random_State and so on.

3.4 Performance Measurement

Five measures were used to compare the performances of the machine learning techniques. They are : Precision, Recall, f-measure, Accuracy and AUC(Area under curve)

$$Recall = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$F1\,Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

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

TP: It shows predictive is positive and it is normally true
TN: It shows predictive is negative and it is normally true
FP: It shows predictive is positive and it is normally false
FN: It shows predictive is negative and it is false.

AUC: This value is the area under the Receiver Operator Characteristic(ROC) curve, which is a ratio between 0 and 1, with 1 indicating a perfect classifier and 0.5 indicating a bad model, which is equivalent to a random classification.

4 RESULTS AND ANALYSIS

We performed data preprocessing using python 3.7 and model training and prediction using apache spark in scala 3.2.1. Figure 8 shows how the memory usage increases and eventually falls short for model training and prediction using python in google colab. This help us to understand the need of using distributed processing using spark. Table 2 shows the results of previous most recent work. Notable among them is model with Bayesian Optimization on Random Forest (BO-RF) with F1-score of 56%. Also a Deep neural network autoencoder model was in [10] which gives accuracy of 76%. We have done prediction using 3 models, Logistic regression, KNN and decision trees. Table 3 shows comparitive performances of each model and ensemble learning model which is developed using min-max technique of predictions from each of these three models. Table 3 shows accuracy and f1-score of each proposed model. We can compare results from table 2 and table 3 which shows our model performs better than existing state of art model. We can compare KNN in both tables and we also observer how our data preprocessing techniques and selected features have improved the results further.

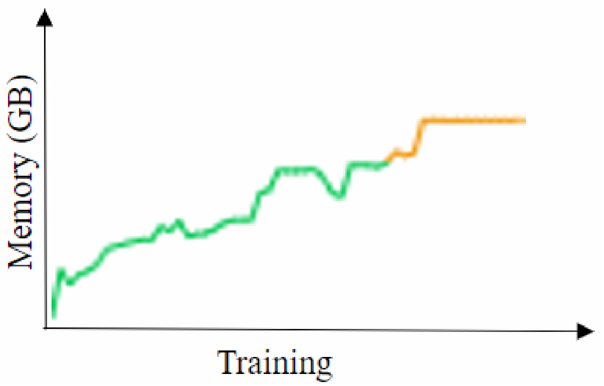


Fig. 8. Memory Usage for training and prediction using python exceeding the memory in google colab (2GB).

Model	Accuracy	Precision	Recall	F1 Score	AUC
ANN [9]		0.47	0.39	0.41	0.9115
KNN [9]		0.46	0.28	0.29	0.616
SVM [9]		0.42	0.3	0.32	0.8938
RF [9]		0.67	0.45	0.51	0.9481
BO-RF [9]		0.64	0.51	0.56	0.956
DNN [10]	0.76				

Table 2. Comparison results of previous work as per literature survey

Model	Accuracy	f1-score
Logistic Regression	0.79	0.81
KNN	0.82	0.87
Decision Trees	0.80	0.86
Ensemble Learning	0.87	0.91

Table 3. Confusion Matrix Report

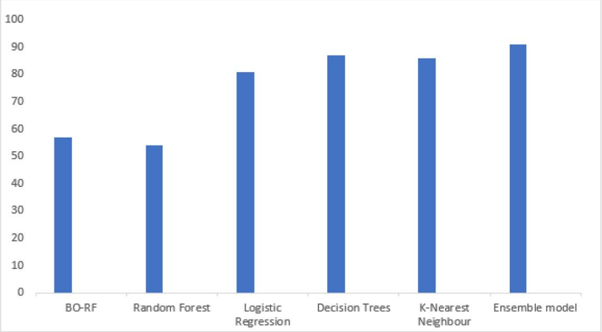


Fig. 9. F1 score comparison according to table 3 and table 2

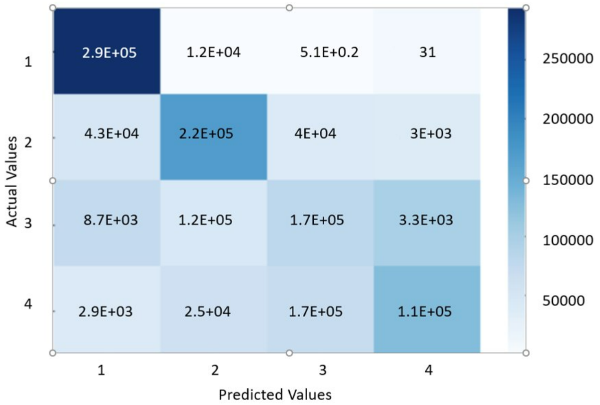


Fig. 10. Confusion Matrix for results with Ensemble learning

5 CONCLUSION AND FUTURE WORK

Traffic accident severity prediction is important for accident management. With the precise prediction of this severity, traffic operators can deploy timely measures to reduce the side effects of the accident. In this study we proposed three machine learning algorithms to build classifiers that are reliable in predicting the accident zones. This includes Logistic Regression(LR), KNN and Decision Trees(DT). We also solved the class imbalance problem of the dataset by using the SMOTE technique. In this study we proposed Ensemble learning algorithms to build classifiers that are reliable in predicting the accident zones. This includes Logistic Regression(LR), Decision Trees(DT) and KNN. We used LR to train the model and predict the results. We also improved the class imbalance problem of the dataset by using the SMOTE technique. We have implemented decision trees in training the model and formed Ensemble learning method which is a combination of LR, DT and KNN. Ensemble technique provides us the advantage of giving better performance by combining the predictions from multiple models. This has been implemented and we achieved F1 score of 91%. As part of our future work, we plan to extend our model to predict accidents to other applications.Improve the model performance to predict more accurately.Develop an API so that this model could be used by applications like Google Maps to help predict accidents and generate notifications for accidents

REFERENCES

[1] Christos Katrakazas , Member, IEEE, Mohammed Quddus, and Wen-Hua Chen, Senior Member, IEEE "A Simulation Study of Predicting Real-Time Conflict-Prone Traffic Conditions" https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8171202

[2] Le yu, Bowen Du,Xiao Hu, Leilei Sun, Weifeng Lv, Runhe Huang "Traffic Accident Prediction Based on Deep Spatio-Temporal Analysis" https://ieeexplore.ieee.org/document/9060300

[3] Hyoungwoo Lee, Jaegul Choo "Data analysis and processing for spatio-temporal forecasting" https://ieeexplore.ieee.org/document/9346316

[4] https://www.kaggle.com/sobhanmoosavi/us-accidents

[5] Xiao-Ling Xia, Bing Nan, Cui Xu "Real-Time Traffic Accident Severity Prediction Using Data Mining Technologies" https://ieeexplore.ieee.org/document/8842686

[6] Rene Richard, Suprio Ray "A tale of two cities: Analyzing road accidents with big spatial data" https://ieeexplore.ieee.org/document/8258334

[7] Li Liu, Ling Shao, Ke Lu "Learning Spatio-Temporal Representations for Action Recognition: A Genetic Programming Approach" https://ieeexplore.ieee.org/document/7042326

[8] Chawla, Nitesh V., Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. "SMOTE: synthetic minority over-sampling technique." Journal of artificial intelligence research 16 (2002): 321-357.

[9] Yan, Miaomiao, and Yindong Shen. "Traffic Accident Severity Prediction Based on Random Forest." Sustainability 14, no. 3 (2022): 1729.

[10] Bibb, Meghan, Pablo Rivas, and Mahee Tayba. "Predicting Traffic Accident Severity with Deep Neural Networks."

[11] M. Chong, A. Abraham, M. Paprzycki, "Traffic accident data mining using machine learning paradigms", Fourth International Conference on Intelligent Systems Design and Applications (ISDA'04), Hungary, 2004, pp. 415- 420.

[12] [12] S. Krishnaveni and M. Hemalatha, "A perspective analysis of traffic accident using data mining 'techniques," International Journal of Computer Applications, vol. 23, no. 7, 2011, pp.40-48.

[13] T. Beshah and S. Hill, "Mining road traffic accident data to improve safety: role of road-related factors on accident severity in Ethiopia," AAAI Spring Symposium, 2010.

[14] G. Chen, Z. Zhang, R. Qian, R. A. Tarefder, and Z. Tian, "Investigating Driver Injury Severity Patterns in Rollover Crashes Using Support Vector Machine Models," Accident Analysis and Prevention, vol. 90, 2016, pp. 128–139.

[15] M. Chong, A. Abraham, and M. Paprzycki, "Traffic accident data mining using machine learning paradigms," in Fourth International Conference on Intelligent Systems Design and Applications (ISDA'04), Hungary, 2018, pp. 415–420.

[16] Hébert, A.; Guédon, T.; Glatard, T.; Jaumard, B. High-Resolution Road Vehicle Collision Prediction for the City of Montreal. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019.

[17] H. Al Najada and I. Mahgoub, "Big vehicular traffic data mining: Towards accident and congestion prevention," in 2016 International Wireless Communications and Mobile Computing Conference (IWCMC). IEEE, 2017, pp. 256–261.

[18] H. Al Najada and I. Mahgoub, "Anticipation and alert system of congestion and accidents in vanet using big data analysis for intelligent transportation systems," in 2016 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2016, pp. 1–8.

[19] A. Elfar, A. Talebpour, and H. S. Mahmassani, "Machine learning approach to short-term traffic congestion prediction in a connected environment," Transportation Research Record, vol. 2672, no. 45, pp. 185–195, 2018.

[20] 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.

[21] Dong. C, Shao. C, Li. J, and Xiong. Z, "An improved deep learning model for traffic crash prediction," Journal of Advanced Transportation, 2018,pp.1-13.

[22] Roshandel, S.; Zheng, Z.; Washington, S. Impact of real-time traffic characteristics on freeway crash occurrence: Systematic review and meta-analysis. Accid. Anal. Prev. 2015, 79, 198–211.

[23] Elvik, R. A Survey of Operational Definitions of Hazardous Road Locations in Some European Countries; Accident Analysis Prevention: Amsterdam, The Netherlands, 2008.

[24] Yuan, Z.; Zhou, X.; Yang, T.; Tamerius, J. 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, 13–17 August 2017.

[25] Sisodia, D.; Singh, L.; Sisodia, S. Clustering Techniques: A Brief Survey of Different Clustering Algorithms. Int. J. Latest Trends Eng. Technol. 2012, 1, 82–87.

[26] R. Genuer, J.-M. Poggi, and C. Tuleau-Malot, "Variable selection using random forests," Pattern Recognition Letters, vol. 31, no. 14, pp. 2225–2236, 2010.

[27] B. García de Soto, A. Bumbacher, M. Deublein, and B. T. Adey, "Predicting road traffic accidents using artificial neural network models," Infrastructure Asset Management, vol. 5, no. 4, pp. 132–144, 2018.

[28] Z. Pu, Z. Li, Y. Jiang, and Y. Wang, "Full bayesian before-after analysis of safety effects of variable speed limit system," IEEE Transactions on Intelligent Transportation Systems, vol. 99, pp. 1–13, 2020.

[29] Z. Yan, X. Lu, and W. Hu, "Analysis of factors affecting traffic accident severity based on heteroskedasticity ordinal Logit," in Proceedings of the Sixth International Conference on Transportation EngineeringICTE 2019, pp. 422–435, American Society of Civil Engineers Reston, Chengdu, China, September 2020.

[30] K. Li, D. Qian, S. Huang, and X. Liang, "Analysis of traffic accidents on highways using latent class clustering," in Proceedings of the 19th COTA International Conference of Transportation Professionals-CICTP, pp. 1800–1810, Nanjing, China, July 2016.

[31] Abegaz T, Gebremedhin S (2019) Magnitude of road traffic accident related injuries and fatalities in Ethiopia. PLoS one 14(1):e0202240

[32] Gu X, Li T, Wang Y, Zhang L, Wang Y, Yao J (2018) Traffic fatalities prediction using support vector machine with hybrid particle swarm optimization. J Algorithms Comput Technol 12(1):20–29

[33] Hartigan JA, Wong MA (1979) Algorithm as 136: a k-means clustering algorithm. J R Stat Soc Ser C (Appl Stat) 28(1):100–108

[34] Persson A (2008) Road traffic accidents in Ethiopia: magnitude, causes and possible interventions. Adv Transp Stud 15:5–16

[35] Sarkar S, Vinay S, Raj R, Maiti J, Mitra P (2019) Application of optimized machine learning techniques for prediction of occupational accidents. Comput Oper Res 106:210–224