Traffic Accident Analysis Using Machine Learning Paradigms

Miao Chong[1], Ajith Abraham[2] and Marcin Paprzycki[1, 3]
[1]Computer Science Department, Oklahoma State University, USA, marcin@cs.okstate.edu
[2]School of Computer Science and Engineering, Chung-Ang University, Korea, ajith.abraham@ieee.org
[3]Computer Science, SWPS, Warszawa, Poland

Keywords: traffic accident, data mining, machine learning, hybrid system, decision trees, support vector machine

Received: December 20, 2004

Engineers and researchers in the automobile industry have tried to design and build safer automobiles, but traffic accidents are unavoidable. Patterns involved in dangerous crashes could be detected if we develop accurate prediction models capable of automatic classification of type of injury severity of various traffic accidents. These behavioral and roadway accident patterns can be useful to develop traffic safety control policies. We believe that to obtain the greatest possible accident reduction effects with limited budgetary resources, it is important that measures be based on scientific and objective surveys of the causes of accidents and severity of injuries. This paper summarizes the performance of four machine learning paradigms applied to modeling the severity of injury that occurred during traffic accidents. We considered neural networks trained using hybrid learning approaches, support vector machines, decision trees and a concurrent hybrid model involving decision trees and neural networks. Experiment results reveal that among the machine learning paradigms considered the hybrid decision tree-neural network approach outperformed the individual approaches.

1 Introduction

The costs of fatalities and injuries due to traffic accidents have a great impact on the society. In recent years, researchers have paid increasing attention to determining factors that significantly affect severity of driver injuries caused by traffic accidents [29][30]. There are several approaches that researchers have employed to study this problem. These include neural network, nesting logic formulation, log-linear model, fuzzy ART maps and so on.

Applying data mining techniques to model traffic accident data records can help to understand the characteristics of drivers' behaviour, roadway condition and weather condition that were causally connected with different injury severity. This can help decision makers to formulate better traffic safety control policies. Roh et al. [22] illustrated how statistical methods based on directed graphs, constructed over data for the recent period, may be useful in modelling traffic fatalities by comparing models specified using directed graphs to a model, based on out-of-sample forecasts, originally developed by Peltzman [23]. The directed graphs model outperformed Peltzman's model in root mean squared forecast error.

Ossenbruggen et al. [24] used a logistic regression model to identify statistically significant factors that predict the probabilities of crashes and injury crashes aiming at using these models to perform a risk assessment of a given region. These models were functions of factors that describe a site by its land use activity, roadside design, use of traffic control devices and traffic exposure. Their study illustrated that village sites are less hazardous than residential and shopping sites. Abdalla et al. [25] studied the relationship between casualty frequencies and the distance of the accidents from the zones of residence. As might have been anticipated, the casualty frequencies were higher nearer to the zones of residence, possibly due to higher exposure. The study revealed that the casualty rates amongst residents from areas classified as relatively deprived were significantly higher than those from relatively affluent areas.

Miaou et al. [26] studied the statistical properties of four regression models: two conventional linear regression models and two Poisson regression models in terms of their ability to model vehicle accidents and highway geometric design relationships. Roadway and truck accident data from the Highway Safety Information System (HSIS) have been employed to illustrate the use and the limitations of these models. It was demonstrated that the conventional linear regression models lack the distributional property to describe adequately random, discrete, nonnegative, and typically sporadic vehicle accident events on the road. The Poisson regression models, on the other hand, possess most of the desirable statistical properties in developing the relationships.

Abdelwahab et al. studied the 1997 accident data for the Central Florida area [2]. The analysis focused on vehicle accidents that occurred at signalized intersections. The injury severity was divided into three classes: no injury, possible injury and disabling injury. They compared the performance of Multi-layered

Perceptron (MLP) and Fuzzy ARTMAP, and found that the MLP classification accuracy is higher than the Fuzzy ARTMAP. Levenberg-Marquardt algorithm was used for the MLP training and achieved 65.6 and 60.4 percent classification accuracy for the training and testing phases, respectively. The Fuzzy ARTMAP achieved a classification accuracy of 56.1 percent.

Yang et al. used neural network approach to detect safer driving patterns that have less chances of causing death and injury when a car crash occurs [17]. They performed the Cramer's V Coefficient test [18] to identify significant variables that cause injury to reduce the dimensions of the data. Then, they applied data transformation method with a frequency-based scheme to transform categorical codes into numerical values. They used the Critical Analysis Reporting Environment (CARE) system, which was developed at the University of Alabama, using a Backpropagation (BP) neural network. They used the 1997 Alabama interstate alcohol-related data, and further studied the weights on the trained network to obtain a set of controllable cause variables that are likely causing the injury during a crash. The target variable in their study had two classes: injury and non-injury, in which injury class included fatalities. They found that by controlling a single variable (such as the driving speed, or the light conditions) they potentially could reduce fatalities and injuries by up to 40%.

Sohn et al. applied data fusion, ensemble and clustering to improve the accuracy of individual classifiers for two categories of severity (bodily injury and property damage) of road traffic accidents [15]. The individual classifiers used were neural network and decision tree. They applied a clustering algorithm to the dataset to divide it into subsets, and then used each subset of data to train the classifiers. They found that classification based on clustering works better if the variation in observations is relatively large as in Korean road traffic accident data.

Mussone et al. used neural networks to analyze vehicle accident that occurred at intersections in Milan, Italy [12]. They chose feed-forward MLP using BP learning. The model had 10 input nodes for eight variables (day or night, traffic flows circulating in the intersection, number of virtual conflict points, number of real conflict points, type of intersection, accident type, road surface condition, and weather conditions). The output node was called an accident index and was calculated as the ratio between the number of accidents for a given intersection and the number of accidents at the most dangerous intersection. Results showed that the highest accident index for running over of pedestrian occurs at non-signalized intersections at nighttime.

Dia et al. used real-world data for developing a multi-layered MLP neural network freeway incident detection model [5]. They compared the performance of the neural network model and the incident detection model in operation on Melbourne's freeways. Results showed that neural network model could provide faster and more reliable incident detection over the model that was in operation. They also found that failure to provide speed data at a station could significantly deteriorate model performance within that section of the freeway.

Shankar et al. applied a nested logic formulation for estimating accident severity likelihood conditioned on the occurrence of an accident [14]. They found that there is a greater probability of evident injury or disabling injury/fatality relative to no evident injury if at least one driver did not use a restraint system at the time of the accident.

Kim et al. developed a log-linear model to clarify the role of driver characteristics and behaviors in the causal sequence leading to more severe injuries. They found that alcohol or drug use and lack of seat belt use greatly increase the odds of more severe crashes and injuries [8].

Abdel-Aty et al. used the Fatality Analysis Reporting System (FARS) crash databases covering the period of 1975-2000 to analyze the effect of the increasing number of Light Truck Vehicle (LTV) registrations on fatal angle collision trends in the US [1]. They investigated the number of annual fatalities that resulted from angle collisions as well as collision configuration (car-car, car-LTV, LTV-car, and LTV-LTV). Time series modeling results showed that fatalities in angle collisions will increase in the next 10 years, and that they are affected by the expected overall increase of the percentage of LTVs in traffic.

Bedard et al. applied a multivariate logistic regression to determine the independent contribution of driver, crash, and vehicle characteristics to drivers' fatality risk [3]. They found that increasing seatbelt use, reducing speed, and reducing the number and severity of driver-side impacts might prevent fatalities. Evanco conducted a multivariate population-based statistical analysis to determine the relationship between fatalities and accident notification times [6]. The analysis demonstrated that accident notification time is an important determinant of the number of fatalities for accidents on rural roadways.

Ossiander et al. used Poisson regression to analyze the association between the fatal crash rate (fatal crashes per vehicle mile traveled) and the speed limit increase [13]. They found that the speed limit increase was associated with a higher fatal crash rate and more deaths on freeways in Washington State.

Finally, researchers studied the relationship between drivers' age, gender, vehicle mass, impact speed or driving speed measure with fatalities and the results of their work can be found in [4, 9, 10, 11, 16].

This paper investigates application of neural networks, decision trees and a hybrid combination of decision tree and neural network to build models that could predict injury severity. The remaining parts of the paper are organized as follows. In Section 2, more details about the problem and the pre-processing of data to be used are presented, followed, in Section 3, by a short description the different machine learning paradigms used. Performance analysis is presented in Section 4 and finally some discussions and conclusions are given towards the end.

2 Accident Data Set

A. Description of the Dataset

This study used data from the National Automotive Sampling System (NASS) General Estimates System (GES) [21]. The GES datasets are intended to be a nationally representative probability samples from the annual estimated 6.4 million accident reports in the United States. The initial dataset for the study contained traffic accident records from 1995 to 2000, a total number of 417,670 cases. According to the variable definitions for the GES dataset, this dataset has drivers' records only and does not include passengers' information. The total set includes labels of year, month, region, primary sampling unit, the number describing the police jurisdiction, case number, person number, vehicle number, vehicle make and model; inputs of drivers' age, gender, alcohol usage, restraint system, eject, vehicle body type, vehicle age, vehicle role, initial point of impact, manner of collision, rollover, roadway surface condition, light condition, travel speed, speed limit and the output injury severity. The injury severity has five classes: *no injury*, *possible injury*, *non-incapacitating injury*, *incapacitating injury*, and *fatal injury*. In the original dataset, 70.18% of the cases have output of no injury, 16.07% of the cases have output of possible injury, 9.48% of the cases have output of non-incapacitating injury, 4.02% of the cases have output of incapacitating injury, and 0.25% of the cases have fatal injury.

Our task was to develop machine learning based intelligent models that could accurately classify the severity of injuries (5 categories). This can in turn lead to greater understanding of the relationship between the factors of driver, vehicle, roadway, and environment and driver injury severity. Accurate results of such data analysis could provide crucial information for the road accident prevention policy. The records in the dataset are input/output pairs with each record have an associated output. The output variable, the injury severity, is categorical and (as described above) has five classes. A supervised learning algorithm will try to map an input vector to the desired output class.

B. Data Preparation

When the input and output variables are considered there are no conflicts between the attributes since each variable represents its own characteristics. Variables are already categorized and represented by numbers. The manner in which the collision occurred has 7 categories: non-collision, rear-end, head-on, rear-to-rear, angle, sideswipe same direction, and sideswipe opposite direction. For these 7 categories the distribution of the fatal injury is as follows: 0.56% for non collision, 0.08% for rear-end collision, 1.54% for head-on collision, 0.00% for rear-to-rear collision, 0.20% for angle collision, 0.08% for sideswipe same direction collision, 0.49% for sideswipe opposite direction collision. Since *head-on* collision has the highest percent of fatal injury; therefore, the dataset was narrowed down to head-on collision only. Head-on collision has a total of 10,386 records, where 160 records show the result as a *fatal injury*; all of these 160 records have the initial point of impact categorized as *front*.

The initial point of impact has 9 categories: no damage/non-collision, front, right side, left side, back, front right corner, front left corner, back right corner, back left corner. The head-on collision with *front impact* has 10,251 records; this is 98.70% of the 10,386 head-on collision records. We have therefore decided to focus on front impact only and removed the remaining 135 records. Travel speed and speed limit were not used in the model because in the dataset there are too many records with unknown value. Specifically, for 67.68% of records the travel speed during accident and local speed limit were unknown. This means that the remaining input variables were: drivers' age, gender, alcohol usage, restraint system, eject, vehicle body type, vehicle role, vehicle age, rollover, road surface condition, light condition. Table 1 summarizes the driver injury severity distribution for head-on collision and front impact point dataset. From Table 1, it is immediately evident that the alcohol usage and not using seat belt, ejection of driver, driver's age (>65), vehicle rollover, and lighting condition can be associated with higher percentages of fatal injury, incapacitating injury and non-incapacitating injury.

There are only single vehicles with ages 37, 41, 46 and 56 years reported in the dataset and therefore these four records were deleted from the dataset (since they were clear outliers). After the preprocessing was completed, the final dataset used for modeling had 10,247 records. There were 5,171 (50.46%) records with no injury, 2138 (20.86%) records with possible injury, 1721 (16.80%) records with non-incapacitating injury, 1057 (10.32%) records with incapacitating injury, and 160 (1.56%) records with fatal injury. We have separated each output class and used one-against-all approach. This approach selects one output class to be the positive class, and all the other classes are combined to be the negative class. We set the output value of the positive class to 1, and the (combined) negative classes to 0. We divided the datasets randomly into 60%, 20%, and 20% for training, cross-validation, and testing respectively.

To make sure that our data preparation is valid, we have checked the correctness of attribute selection. There are several attribute selection techniques to find a minimum set of attributes so that the resulting probability distribution of the data classes is as close as possible to the original distribution of all attributes. To determine the best and worst attributes, we used the chi-squared (χ^2) test to determine the dependence of input and output variables. The χ^2 test indicated that all the variables are significant (p-value < 0.05).

3. Machine Learning Paradigms

A. Artificial Neural Networks Using Hybrid Learning

A Multilayer Perceptron (MLP) is a feed forward neural network with one or more hidden layers.

Table 1: Driver injury severity distribution

Factor	No Injury	Pos injury	Non-incapacitating	Incapacitating	Fatal	Total
Age						
0 (24&under)	1629(52.80%)	608(19.71%)	505(16.37%)	307(9.95%)	36(1.17%)	3085
1 (25-64)	3171(49.88%)	1362(21.43%)	1075(16.91%)	654(10.29%)	95(1.49%)	6357
2 (65+)	373(46.11%)	168(20.77%)	143(17.68%)	96(11.87%)	29(3.58%)	809
Gender						
0 (Female)	1749(41.95%)	1072(25.71%)	778(18.66%)	507(12.16%)	63(1.51%)	4169
1 (Male)	3424(56.30%)	1066(17.53%)	945(15.54%)	550(9.04%)	97(1.59%)	6082
Eject						
0 (No Eject)	5171(50.55%)	2137(20.89%)	1719(16.80%)	1047(10.23%)	156(1.52%)	10230
1 (Eject)	2(9.52%)	1(4.76%)	4(19.05%)	10(47.62%)	4(19.05%)	21
Alcohol						
0 (No Alcohol)	4997(51.35%)	2067(21.24%)	1600(16.44%)	935(9.61%)	133(1.37%)	9732
1 (Alcohol)	176(33.91%)	71(13.68%)	123(23.70%)	122(23.51%)	27(5.20%)	519
Restraining System						
0 (Not Used)	337(27.44%)	193(15.72%)	336(27.36%)	283(23.05%)	79(6.43%)	1228
1 (Used)	4836(53.60%)	1945(21.56%)	1387(15.37%)	774(8.58%)	81(0.90%)	9023
Body Type						
0 (cars)	3408(47.49%)	1600(22.30%)	1272(17.73%)	780(10.87%)	116(1.62%)	7176
1 (SUV &Van)	747(56.59%)	259(19.62%)	189(14.32%)	111(8.41%)	14(1.06%)	1320
2 (Truck)	1018(58.01%)	279(15.90%)	262(14.93%)	166(9.46%)	30(1.71%)	1755
Vehicle Role						
1 (Striking)	4742(49.86%)	2011(21.15%)	1636(17.20%)	970(10.20%)	151(1.59%)	9510
2 (Struck)	261(72.70%)	54(15.04%)	29(8.08%)	15(4.18%)	0(0%)	359
3 (Both)	170(44.50%)	73(19.11%)	58(15.18%)	72(18.85%)	9(2.36%)	382
Rollover						
0 (No-rollover)	5069(50.78%)	2123(20.85%)	1699(16.69%)	1037(10.19%)	152(1.49%)	10180
1 (Rollover)	4(5.63%)	15(21.13%)	24(33.80%)	20(28.17%)	8(11.27%)	71
Road Surface Condition						
0 (Dry)	3467(49.97%)	1404(20.24%)	1190(17.15%)	750(10.81%)	127(1.83%)	6938
1 (Slippery)	1706(51.49%)	734(22.16%)	533 (16.09%)	307(9.27%)	33(1.00%)	3313
Light Condition						
0 (Daylight)	3613(51.18%)	1487(21.06%)	1174(16.63%)	688(9.75%)	98(1.39%)	7060
1(Partial dark)	1139(52.71%)	465(21.52%)	348(16.10%)	186(8.61%)	23(1.06%)	2161
2 (Dark)	421(40.87%)	186(18.06%)	201(19.51%)	183(17.77%)	39(3.79%)	1030

The network consists of an input layer of source neurons, at least one hidden layer of computational neurons, and an output layer of computational neurons. The input layer accepts input signals and redistributes these signals to all neurons in the hidden layer. The output layer accepts a stimulus pattern from the hidden layer and establishes the output pattern of the entire network. The MLP neural networks training phase works as follows: given a collection of training data $\{x_1(p), d_1(p)\}$, …, $\{x_i(p), d_i(p)\}$, …, $\{x_n(p), d_n(p)\}$, the objective is to obtain a set of weights that makes almost all the tuples in the training data classified correctly, or in other words, is to map $\{x_1(p)$ to $d_1(p)\}$, …, $\{x_i(p)$ to $d_i(p)\}$, and eventually $\{x_n(p)$ to $d_n(p)\}$. The algorithm starts with initializing all the weights (w) and threshold (θ) levels of the network to small random numbers. Then calculate the actual output of the neurons in the hidden layer as:

$$y_i(p) = f\,[\textstyle\sum_{(i=1 \text{ to } n)} x_i(p) * w_{ij}(p) - \theta_j],$$

where n is the number of inputs of neuron j in the hidden layer. Next calculate the actual outputs of the neurons in the output layer as:

$$y_k(p) = f\,[\textstyle\sum_{(j=1 \text{ to } m)} x_{jk}(p) * w_{jk}(p) - \theta_k],$$

where m is the number of inputs of neuron k in the output layer. The weight training is to update the weights using the Backpropagation (BP) learning method with the error function:

$E\ (\mathrm{w}) = \sum_{(p=1 \text{ to } PT)} \sum_{(i=1 \text{ to } l)} [d_i(p) - y_i(p)]^2$,

where

E (w) = error function to be minimized,

w = weight vector,

PT = number of training patterns,

l = number of output neurons,

$d_i(p)$ = desired output of neuron I when pattern p is introduced to the MLP, and

$y_i(p)$ = actual output of the neuron I when pattern p is introduced to the MLP. The objective of weight training is to change the weight vector w so that the error function is minimized. By minimizing the error function, the actual output is driven closer to the desired output.

Empirical research [19] has shown that the BP used for training neural networks has the following problems:

- BP often gets trapped in a local minimum mainly because of the random initialization of weights.
- BP usually generalizes quite well to detect the global features of the input but after prolonged training the network will start to recognize individual input/output pair rather than settling for weights that generally describe the mapping for the whole training set.

The second popular training algorithm for neural networks is Scaled Conjugate Gradient Algorithm (SCGA). Moller [20] introduced it as a way of avoiding the complicated line search procedure of conventional conjugate gradient algorithm (CGA). According to the SCGA, the Hessian matrix is approximated by

$$E''(w_k)p_k = \frac{E'(w_k + \sigma_k p_k) - E'(w_k)}{\sigma_k} + \lambda_k p_k$$

where E' and E'' are the first and second derivative information of global error function $E\ (w_k)$. The other terms p_k, σ_k and λ_k represent the weights, search direction, parameter controlling the change in weight for the second derivative approximation and parameter for regulating the indefiniteness of the Hessian. In order to obtain a good, quadratic, approximation of E, a mechanism to raise and lower λ_k is needed when the Hessian is positive definite. Detailed step-by-step description can be found in [20].

In order to minimize the above-mentioned problems resulting from the BP training, we used a combination of BP and SCG for training.

B. Decision Trees

Decision trees are well-known algorithm for classification problems. The Classification and Regression Trees (CART) model consists of a hierarchy of univariate binary decisions. Each internal node in the tree specifies a binary test on a single variable, branch represents an outcome of the test, each leaf node represent class labels or class distribution. CART operates by choosing the best variable for splitting the data into two groups at the root node, partitioning the data into two disjoint branches in such a way that the class labels in each branch are as homogeneous as possible, and then splitting is recursively applied to each branch, and so forth.

If a dataset T contains examples from n classes, gini index, $gini(T)$ is defined as: $gini\ (T) = 1 - \sum_{j=1 \text{ to } n} p_j$^2, where p_j is the relative frequency of class j in T [31]. If dataset T is split into two subsets T_1 and T_2 with sizes N_1 and N_2, the gini index of the split data contains examples from n classes, the gini index $gini(T)$ is defined as:

$gini_{\text{split}}\ (T) = N_1/N\ gini(T_1) + N_2/N\ gini(T_2)$.

CART exhaustively searches for univariate splits. The attribute provides the smallest $gini_{\text{split}}\ (T)$ is chosen to split the node. CART recursively expands the tree from a root node, and then gradually prunes back the large tree. The advantage of a decision tree is the extraction of classification rules from trees that is very straightforward. More precisely, a decision tree can represent the knowledge in the form of if-then rules; one rule is created for each path from the root to a leaf node.

C. Support Vector Machines

Support Vector Machine (SVM) is based on statistical learning theory [28] . SVMs have been successfully applied to a number of applications ranging from handwriting recognition, intrusion detection in computer networks, and text categorization to image classification, breast cancer diagnosis and prognosis and bioinformatics. SVM involves two key techniques, one is the mathematical programming and the other is kernel functions. Here, parameters are found by solving a quadratic programming problem with linear equality and inequality constraints; rather than by solving a non-convex, unconstrained optimization problem. SVMs are kernel-based learning algorithms in which only a fraction of the training examples are used in the solution (these are called the support vectors), and where the objective of learning is to maximize a margin around the decision surface. The flexibility of kernel functions allows the SVM to search a wide variety of hypothesis spaces. The basic idea of applying SVMs to pattern classification can be stated briefly as: first map the input vectors into one feature space (possible with a higher dimension), either linearly or nonlinearly, whichever is relevant to the selection of the kernel function; then within the feature space, seek an optimized linear division, i.e. construct a hyperplane which separates two classes.

For a set of n training examples (x_i, y_i), where $x_i \in \boldsymbol{R}^d$ and $y_i \in \{-1, +1\}$, suppose there is a hyperplane, which separates the positive from the negative examples. The points x which lie on the hyperplane (H_0) satisfy $w \cdot x + b = 0$, the algorithm finds this hyperplane (H_0) and other two hyperplanes (H_1, H_2) parallel and equidistant to H_0,

H_1: $w \cdot x_i + b = 1$, H_2: $w \cdot x_i + b = -1$,

H_1 and H_2 are parallel and no training points fall between them. Support vector algorithm looks for the separating hyperplane and maximizes the distance between H_1 and H_2. So there will be some positive examples on H_1 and some negative examples on H_2. These examples are called support vectors. The distance between H_1 and H_2 is $2/||w||$, in order to maximize the distance, we should minimize $||w|| = w^T w$, subject to constraints $y_i (w \cdot x_i + b) >= 1, \forall_i$

Introducing Lagrangian multipliers $\alpha_1, \alpha_2, \ldots, \alpha n >= 0$, the learning task becomes

$L (w, b, \alpha) = ½ w^T w - \sum_{i=1 \text{ to } n} \alpha_I [y_i(w \cdot x_i + b) - 1]$

The above equation is for two classes that are linearly separable. When the two classes are non-linearly separable, SVM can transform the data points to another high dimensional space. Detailed description to the theory of SVMs for pattern recognition can be found in [32].

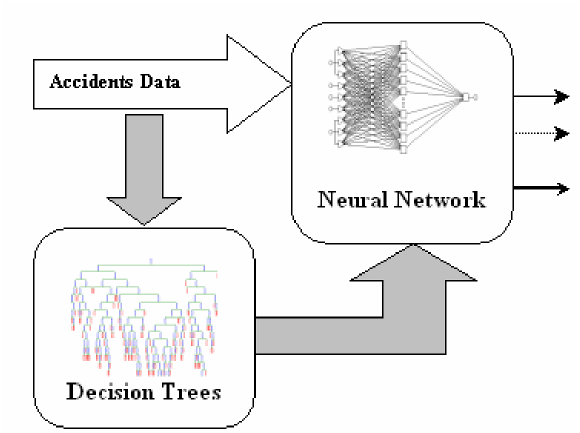

Fig. 1. Hybrid concurrent decision tree-ANN model for accident data

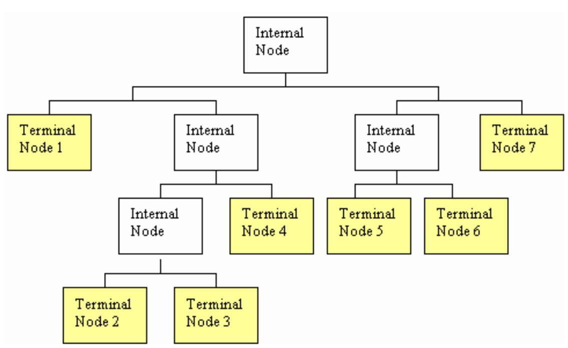

Fig. 2. Decision tree structure

D. Hybrid Decision Tree-ANN (DTANN)

A hybrid intelligent system uses the approach of integrating different learning or decision-making models. Each learning model works in a different manner and exploits different set of features. Integrating different learning models gives better performance than the individual learning or decision-making models by reducing their individual limitations and exploiting their different mechanisms. In a hierarchical hybrid intelligent system each layer provides some new information to the higher level [33]. The overall functioning of the system depends on the correct functionality of all the layers. Figure 1 illustrates the hybrid decision tree-ANN (DTANN) model for predicting drivers' injury severity. We used a concurrent hybrid model where traffic accidents data are fed to the decision tree to generate the node information. Terminal nodes were numbered left to right starting with 1. All the data set records were assigned to one of the terminal nodes, which represented the particular class or subset. The training data together with the node information were supplied for training the ANN. Figure 2 illustrates a decision tree structure with the node numbering. For the hybrid decision tree–ANN, we used the same hybrid learning algorithms and parameters setting as we used for ANN (except for the number of hidden neurons). Experiments were performed with different number of hidden neurons and models were selected with the highest classification accuracy for the output class.

4. Performance Analysis

A. Neural Networks

In the case of neural network based modeling, the hyperbolic activation function was used in the hidden layer and the logistic activation function in the output layer. Models were trained with BP (100 epochs, learning rate 0.01) and SCGA (500 epochs) to minimize the Mean Squared Error (MSE). For each output class, we experimented with different number of hidden neurons, and report the model with highest classification accuracy for the class. From the experiment results, for the no injury class the best model had 65 hidden neurons, and achieved training and testing performance of 63.86% and 60.45% respectively. For the possible injury class, the best model had 65 hidden neurons achieving it's training and testing performance of 59.34% and 57.58% respectively. For the non-incapacitating injury class, the best model had 75 hidden neurons achieving training and testing performance of 58.71% and 56.8% respectively. For the incapacitating injury class, the best model had 60 hidden neurons achieving training and testing performance of 63.40% and 63.36% respectively. Finally, for the fatal injury class, the best model had 45 hidden neurons achieving training and testing performance of 78.61% and 78.17% respectively. These results are the summary of multiple experiments (for variable no of hidden neurons and for a number of attempts with random initial weight distributions resulting in almost exact performance of the trained network) and are presented in Table 2.

B. Decision Trees

We have experimented with a number of setups of decision tree parameters and report the best results obtained for our dataset. We trained each class with Gini goodness of fit measure, the prior class probabilities parameter was set to equal, the stopping option for pruning was misclassification error, the minimum *n* per node was set to 5, the fraction of objects was 0.05, the maximum number of nodes was 1000, the maximum number of levels in the tree was 32, the number of surrogates was 5, we used 10 fold cross-validation, and generated comprehensive results. The cross-validation testing ensured that the patterns found will hold up when applied to new data.

Table 2. Neural network performance

Table 2. Neural network performance **No Injury**			**Possible Injury**			**Non-incapacitating**			**Incapacitating**			**Fatal Injury**		
# neuron s	Accuracy %		# neuron s	Accuracy %		# neuron s	Accuracy %		# neuron s	Accuracy %		# neuron s	Accuracy %	
	Train	Test		Train	Test		Train	Test		Train	Test		Train	Test
60	63.57	59.67	65	59.34	**57.58**	60	57.88	55.25	60	63.4	**63.36**	45	77.26	**75.17**
65	63.86	**60.45**	70	59.56	55.15	65	57.69	54.66	65	62.23	61.32	57	74.78	70.65
70	63.93	60.25	75	58.88	57.29	75	58.71	**56.80**	75	61.06	61.52	65	69.81	69.73
75	64.38	57.43	80	58.39	56.22	80	57.78	54.13	84	63.23	58.41	75	60.19	59.62
80	63.64	58.89	95	60.07	55.93	85	57.83	55.59	90	59.32	59.08	80	74.33	71.77

Table 3: Performance of SVM using radial basis function kernel

	g=0.0001 c=42.8758	g=0.001 c=4.6594	g=0.5 c=0.5	g=1.2 c=0.5	g=1.5 c=2	g=2 c=10	g=0.00001 c=100	g=0.0001 c=100	g=0.001 c=100
No injury									
Class 0	59.76	59.80	57.95	57.65	53.62	54.12	57.34	59.76	60.46
Class 1	60.14	60.14	60.82	55.63	55.73	55.53	62.88	60.14	60.14
Possible injury									
Class 0	100.00	100.00	100.00	99.88	95.33	95.58	100.00	100.00	100.00
Class 1	0.00	0.00	0.00	0.00	3.67	3.42	0.00	0.00	0.00
Non-incapacitating									
Class 0	100.00	100.00	100.00	100.00	97.43	97.49	100.00	100.00	100.00
Class 1	0.00	0.00	0.00	0.00	3.21	2.92	0.00	0.00	0.00
Incapacitating									
Class 0	100.00	100.00	100.00	99.89	98.06	98.11	100.00	100.00	100.00
Class 1	0.00	0.00	0.00	0.00	2.83	2.83	0.00	0.00	0.00
Fatal Injury									
Class 0	100.00	100.00	100.00	100.00	99.95	99.95	100.00	100.00	100.00
Class 1	0.00	0.00	0.00	0.00	3.33	3.33	0.00	0.00	0.00

Table 4. Decision tree performance

Injury Class	**Accuracy (%)**
No Injury	67.54
Possible Injury	64.40
Non-incapacitating Injury	60.37
Incapacitating Injury	71.38
Fatal Injury	89.46

The performance for no injury, possible injury, non-incapacitating injury, incapacitating injury and fatal injury models was 67.54%, 64.39%, 60.37%, 71.38%, and 89.46% respectively. Empirical results including classification matrix are illustrated in Table 4. The developed decision trees are depicted in Figures 3-7. Each of these trees has a completely different structure and number of nodes and leaves. Note, that information stored in leaves of exactly these decision trees has been used in developing the hybrid decision tree – neural network model.

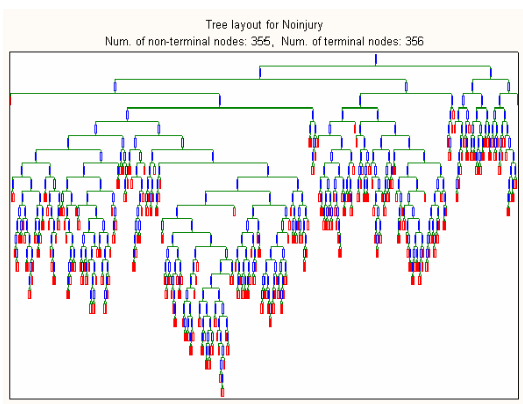

Fig. 3: No injury tree structure

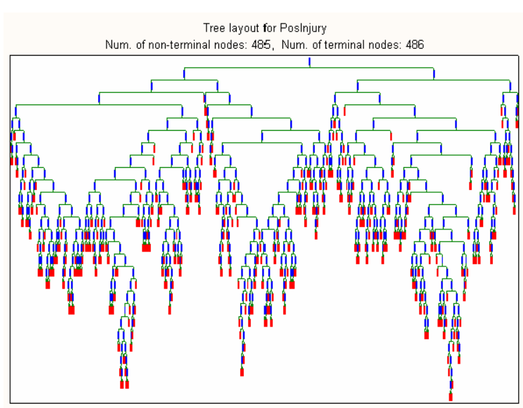

Fig. 4: Possible injury tree structure

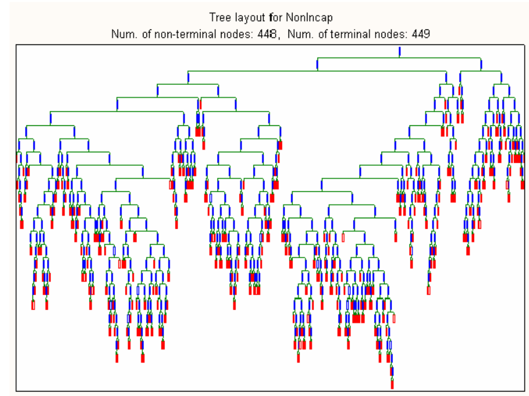

Fig. 5: Non-incapacitating injury tree structure

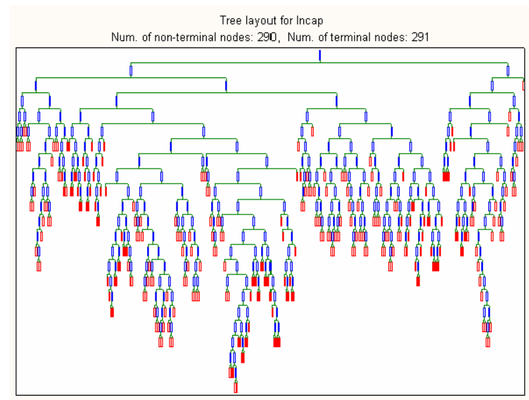

Fig. 6: Incapacitating injury tree structure

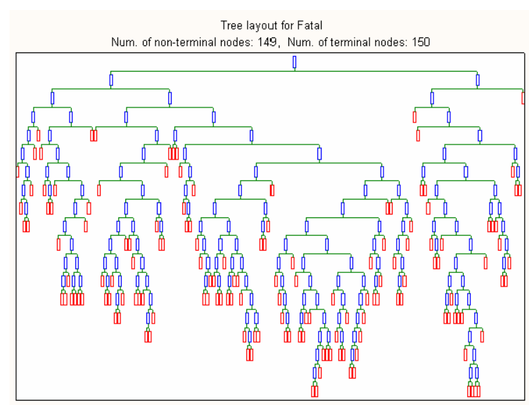

Fig. 7: Fatal injury tree structure

C. Support Vector Machines

In our experiments we used the SVM^{light} [27] and selected the polynomial and radial basis function kernels. For an unknown reason, the polynomial kernel was not successful and hence we only focused on the radial basis function (RBF) kernels. Table 3 illustrates the SVM performance for the different parameter settings and the obtained accuracies for each class.

D. Hybrid DT-ANN Approach

In the case of the hybrid approach, for the no injury class the best model had 70 hidden neurons, with training and testing performance of 83.02% and 65.12% respectively.

For the possible injury class, the best model had 98 hidden neurons with training and testing performance of 74.93% and 63.10% respectively. For the non-incapacitating injury class, the best model had 109 hidden neurons with training and testing performance of 71.88% and 62.24% respectively. For the incapacitating injury class, the best model had 102 hidden neurons, with training and testing performance of 77.95% and 72.63% respectively. Finally, for the fatal injury class, the best model had 76 hidden neurons with training and testing performance of 91.53% and 90.00% respectively. These are the best models out of multiple experiments varying various parameters of the ANN and the decision tree. Empirical results are presented in Table 5 and the final comparison between ANN, DT and DTANN is graphically illustrated in Figure 8. For all the output classes, the hybrid DTANN outperformed the ANN. For non-incapacitating injury, incapacitating injury, and fatal injury classes, the hybrid DTANN outperformed both ANN and DT.

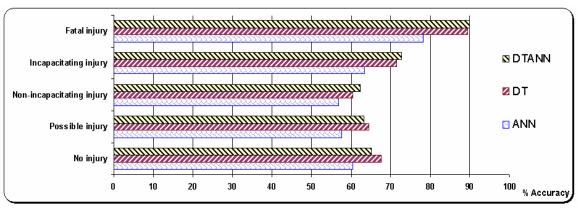

Fig. 8. Performance comparison of the different learning paradigms

Table 5. Test performance of DTANN

Injury type	**% Accuracy**
No injury	65.12
Possible injury	63.10
Non-incapacitating injury	62.24
Incapacitating injury	72.63
Fatal injury	90.00

5. Concluding Remarks

In this paper, we analyzed the GES automobile accident data from 1995 to 2000 and investigated the performance of neural network, decision tree, support vector machines and a hybrid decision tree – neural network based approaches to predicting drivers' injury severity in head-on front impact point collisions. The classification accuracy obtained in our experiments reveals that, for the non-incapacitating injury, the incapacitating injury, and the fatal injury classes, the hybrid approach performed better than neural network, decision trees and support vector machines. For the no injury and the possible injury classes, the hybrid approach performed better than neural network. The no injury and the possible injury classes could be best modeled directly by decision trees.

Past research focused mainly on distinguishing between no-injury and injury (including fatality) classes. We extended the research to possible injury, non-incapacitating injury, incapacitating injury, and fatal injury classes. Our experiments showed that the model for fatal and non-fatal injury performed better than other classes. The ability of predicting fatal and non-fatal injury is very important since drivers' fatality has the highest cost to society economically and socially.

It is well known that one of the very important factors causing different injury level is the actual speed that the vehicle was going when the accident happened. Unfortunately, our dataset doesn't provide enough information on the actual speed since speed for 67.68% of the data records' was unknown. If the speed was available, it is extremely likely that it could have helped to improve the performance of models studied in this paper.

6. References

[1] Abdel-Aty, M., and Abdelwahab, H., Analysis and Prediction of Traffic Fatalities Resulting From Angle Collisions Including the Effect of Vehicles' Configuration and Compatibility. *Accident Analysis and Prevention*, 2003.

[2] Abdelwahab, H. T. and Abdel-Aty, M. A., Development of Artificial Neural Network Models to Predict Driver Injury Severity in Traffic Accidents at Signalized Intersections. *Transportation Research Record 1746*, Paper No. 01-2234.

[3] Bedard, M., Guyatt, G. H., Stones, M. J., & Hireds, J. P., The Independent Contribution of Driver, Crash, and Vehicle Characteristics to Driver Fatalities. *Accident analysis and Prevention*, Vol. 34, pp. 717-727, 2002.

[4] Buzeman, D. G., Viano, D. C., & Lovsund, P., Car Occupant Safety in Frontal Crashes: A Parameter Study of Vehicle Mass, Impact Speed, and Inherent Vehicle Protection. *Accident Analysis and Prevention*, Vol. 30, No. 6, pp. 713-722, 1998.

[5] Dia, H., & Rose, G., Development and Evaluation of Neural Network Freeway Incident Detection Models Using Field Data. *Transportation Research C*, Vol. 5, No. 5, 1997, pp. 313-331.

[6] Evanco, W. M., The Potential Impact of Rural Mayday Systems on Vehicular Crash Fatalities. Accident Analysis and Prevention, Vol. 31, 1999, pp. 455-462.

[7] Hand, D., Mannila, H., & Smyth, P., Principles of Data Mining. The MIT Press, 2001.

[8] Kim, K., Nitz, L., Richardson, J., & Li, L., Personal and Behavioral Predictors of Automobile Crash and

Injury Severity. *Accident Analysis and Prevention*, Vol. 27, No. 4, 1995, pp. 469-481.

[9] Kweon, Y. J., & Kockelman, D. M., Overall Injury Risk to Different Drivers: Combining Exposure, Frequency, and Severity Models. Accident Analysis and Prevention, Vol. 35, 2003, pp. 441-450.

[10] Martin, P. G., Crandall, J. R., & Pilkey, W. D., Injury Trends of Passenger Car Drivers In the USA. Accident Analysis and Prevention, Vol. 32, 2000, pp. 541-557.

[11] Mayhew, D. R., Ferguson, S. A., Desmond, K. J., & Simpson, G. M., Trends In Fatal Crashes Involving Female Drivers, 1975-1998. Accident Analysis and Prevention, Vol. 35, 2003, pp. 407-415.

[12] Mussone, L., Ferrari, A., & Oneta, M., An analysis of urban collisions using an artificial intelligence model. *Accident Analysis and Prevention*, Vol. 31, 1999, pp. 705-718.

[13] Ossiander, E. M., & Cummings, P., Freeway speed limits and Traffic Fatalities in Washington State. Accident Analysis and Prevention, Vol. 34, 2002, pp. 13-18.

[14] Shankar, V., Mannering, F., & Barfield, W., Statistical Analysis of Accident Severity on Rural Freeways. *Accident Analysis and Prevention*, Vol. 28, No. 3, 1996, pp.391-401.

[15] Sohn, S. Y., & Lee, S. H., Data Fusion, Ensemble and Clustering to Improve the Classification Accuracy for the Severity of Road Traffic Accidents in Korea. *Safety Science*, Vol. 4, issue1, February 2003, pp. 1-14.

[16] Tavris, D. R., Kuhn, E. M, & Layde, P. M., Age and Gender Patterns In Motor Vehicle Crash injuries: Improtance of Type of Crash and Occupant Role. Accident Analysis and Prevention, Vol. 33, 2001, pp. 167-172.

[17] Yang, W.T., Chen, H. C., & Brown, D. B., Detecting Safer Driving Patterns By A Neural Network Approach. *ANNIE '99 for the Proceedings of Smart Engineering System Design Neural Network, Evolutionary Programming, Complex Systems and Data Mining*, Vol. 9, pp 839-844, Nov. 1999.

[18] Zembowicz, R. and Zytkow, J. M., 1996. From Contingency Tables to Various Forms of Knowledge in Database. Advances in knowledge Discovery and Data Mining, editors, Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R. AAAI Press/The MIT Press, pp.329-349.

[19] Abraham, A., Meta-Learning Evolutionary Artificial Neural Networks, Neurocomputing Journal, Elsevier Science, Netherlands, Vol. 56c, pp. 1-38, 2004.

[20] Moller, A.F., A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning, Neural Networks, Volume (6), pp. 525-533, 1993.

[21] National Center for Statistics and Analysis http://www-nrd.nhtsa.dot.gov/departments/nrd-30/ncsa/NASS.html

[22] Roh J.W., Bessler D.A. and Gilbert R.F., Traffic fatalities, Peltzman's model, and directed graphs, Accident Analysis & Prevention, Volume 31, Issues 1-2, pp. 55-61, 1998.

[23] Peltzman, S., The effects of automobile safety regulation. Journal of Political Economy 83, pp. 677–725, 1975.

[24] Ossenbruggen, P.J., Pendharkar, J. and Ivan, J., Roadway safety in rural and small urbanized areas. Accid. Anal. Prev. 33 4, pp. 485–498, 2001.

[25] Abdalla, I.M., Robert, R., Derek, B. and McGuicagan, D.R.D., An investigation into the relationships between area social characteristics and road accident casualties. Accid. Anal. Prev. 29 5, pp. 583–593, 1997.

[26] Miaou, S.P. and Harry, L., Modeling vehicle accidents and highway geometric design relationships. Accid. Anal. Prev. 25 6, pp. 689–709, 1993.

[27] SVM^{light}. http://www.cs.cornell.edu/People/tj/svm_light/. Access date: May, 2003.

[28] Vapnik, V. N., The Nature of Statistical Learning Theory. Springer, 1995.

[29] Chong M., Abraham A., Paprzycki M., Traffic Accident Data Mining Using Machine Learning Paradigms, Fourth International Conference on Intelligent Systems Design and Applications (ISDA'04), Hungary, ISBN 9637154302, pp. 415-420, 2004.

[30] Chong M., Abraham A., Paprzycki M., Traffic Accident Analysis Using Decision Trees and Neural Networks, IADIS International Conference on Applied Computing, Portugal, IADIS Press, Nuno Guimarães and Pedro Isaías (Eds.), ISBN: 9729894736, Volume 2, pp. 39-42, 2004.

[31] Eui-Hong (Sam) Han, Shashi Shekhar, Vipin Kumar, M. Ganesh, Jaideep Srivastava, Search Framework for Mining Classification Decision Trees, 1996. umn.edu/dept/users/kumar/dmclass.ps

[32] N. Cristianini and J. Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge University Press, 2000.

[33] Abraham, Intelligent Systems: Architectures and Perspectives, Recent Advances in Intelligent Paradigms and Applications, Abraham A., Jain L. and Kacprzyk J. (Eds.), Studies in Fuzziness and Soft Computing, Springer Verlag Germany, Chapter 1, pp. 1-35, 2002.