A Wide-and-Deep Learning Model of Travel Mode Detection

Jun ZHAO[1], Chenfeng XIONG[2*], Di YANG[1], Ya JI[3], Liang TANG[1], Lei ZHANG[4]

1. Graduate Research Assistant, Department of Civil and Environmental Engineering, University of Maryland, College Park, MD, 20742, USA
2. (Corresponding Author) Assistant Research Professor, Department of Civil and Environmental Engineering, University of Maryland, College Park, MD, 20742, USA. Phone: (301)405-9430; Email: cxiong@umd.edu
3. Research Scientist, Department of Civil and Environmental Engineering, University of Maryland, College Park, MD, 20742, USA
4. Herbert Rabin Distinguished Professor, Department of Civil and Environmental Engineering, University of Maryland, College Park, MD, 20742, USA

ABSTRACT

Artificial intelligence methods are widely used in travel mode detection based on passively collected GPS data. This paper presents a jointly trained single-layer model and deep neural network for travel mode detection. Being “wide” and “deep” at the same time, this model combines the advantages of both types of models to be able to make sufficient generalizations using multi-layer deep learning and capture the exceptions using the wide single-layer model. The model is empirically tested on a GPS dataset collected in the Washington D.C. and Baltimore metropolitan regions. We also innovatively line-up the multimodal transportation network to the GPS trajectories in order to infer the closeness to the nearby rail (both underground and aboveground) and bus lines. To the best knowledge of the authors, this paper is the first to use land use data to infer underground metro modes. The empirical test showcases the superior goodness-of-fit and high precision and recall rates of the proposed wide and deep learning model compared with other benchmark machine learning models.

1. BACKGROUND

Thanks to the vividly growing smartphone industry, passively collected travel data has never been so readily available before. The widespread use of Global Positioning System (GPS)-based technologies, GPS loggers, GPS-enabled phones, etc., provides an innovative but accurate approach to observe and track individuals’ travel behavior. Compared to traditional data collecting activities, without spending abundant time and money, GPS-based technologies play as a trending role in collecting a large amount of accurate spatial and temporal information passively. In addition, the fast development of connected vehicles and autonomous vehicle technologies ensures the continued influx of GPS data. This advent of GPS big data requires technologies and research for processing and utilization to better serve our life. In order to optimally use GPS data, we must be able to infer multiple trip information, such as travel modes and trip purposes.

Mode detection based on GPS raw data drew increasing research attention in the past decade while GPS technology has been widely used to collect large-scale transportation data. Researchers have explored artificial intelligence (AI) methods to cope with mode detection, including Decision Tree (e.g. Stenneth 2011; Zheng, 2008), Neural Network (Gonzalez et al., 2010; Byon, 2014; Yang, 2015), Naïve Bayes and Bayesian Networks (Xiao, et al., 2015), Support Vector Machines (e.g. Zhang, 2011), Random Forest (e.g. Lari, 2015), etc. Wu et al. (2016) have conducted a thorough synthesis of existing studies on this topic. Data, methodologies, and research outcomes are reviewed and compared. Overall, the current practices can detect car mode with high accuracy (with high-definition GPS traces which also draws battery concerns). However, the detection of bus/metro/subway modes is not as satisfying. One possible methodological limitation that could lead to this is the single-layer AI representation. The single-layer neurons or rules often cannot make enough generalization on a high-dimensional problem. To generalize unobserved feature combinations for bus or metro modes, a multi-layer deep neural network (DNN) can perform much efficiently with much fewer nodes in each layer.

In this paper, we present a study employing a jointly trained single-layer model and deep neural network for travel mode detection. This model combines the advantages of both types of models to be able to make sufficient generalizations using a multi-layer DNN and capture the exceptions using the wide single-layer model (Cheng, et al., 2016). We compare its performance with other AI methods, including generalized linear models, decision tree models, support vector machine models and other ensemble model: random forest models, bagging and boosting. In addition to the traditional features used in the literature (e.g. average speed, maximum speed, trip distance, etc.), this study constructed two innovative features based on land use data: the distance to the closest rail line (both underground and aboveground) and the distance to the closest bus line. To the best knowledge of the authors, this paper is the first to use land use data to infer Metro mode that is typically underground.

The remainder of the paper will first introduce the methodology. The benchmark Random Forest model and the proposed wide and deep learning model for travel mode detection will be explained. Then, empirical data passively collected from smartphone GPS loggers provides the training and testing data for the study, as well as the application context. Furthermore, results show superior performance of the wide and deep learning model compared with other models. Goodness-of-fit, precision and recall rates are compared among the models discussed in the end of this paper.

2. MACHINE LEARNING MODELS

2.1 Benchmark Models

Based on the authors' previous studies on rule-based models and decision trees (Xiong and Zhang, 2013; Tang et al., 2016; Tang et al., 2017), the models of generalized linear model, decision tree model, support vector machine and some other ensemble models: random forest and adaptive boosting are selected as the benchmark for this mode detection research. A random forest is an ensemble of decision trees (Ho, 1995), generally trained via the bagging method, or sometimes pasting. In our research, we use the bagging (also know as bootstrap aggregating) method and all the trees share some weights. However, to make the prediction more accurate, we need to pay more attention to the training instances that the predecessor underfitted. This results in new predictors focusing more and more on the hard cases. The technique is called adaptive boosting (adaboost). The adaboost method in this paper also uses a decision tree as the base classifier and change the relative weights based on misclassified instances (Freund et al., 1999).

2.2 A Model of Wide and Deep Learning

In this paper, we explore the mode detection based on a wide and deep learning approach, as illustrated in Figure 1. As previously discussed, this model is capable of generalizing rules and memorizing specific exceptions at the same time. It leads to superior prediction accuracy, compared to stand-alone generalized linear models, decision tree models, DNN models, and the other benchmark models. These models are all trained and fine-tuned using the TensorFlow platform in Python.

GPS trajectory features (trip time, distance, OD distance, avg. speed, and maximum speed) and network features (average distances to the nearest Metro line and bus line) are used in the Wide and Deep model. These features are all continuous and normalized to the range of [0, 1]. More details of the data and variables can be found in the next section. Two hidden layers in the DNN are illustrated in Figure 1, with m neurons and n neurons, respectively. The number of layers and the number of neurons in each layer can be fine-tuned. In the empirical test of this paper, we have used three hidden layers and we have also tested the model using a different number of neurons.

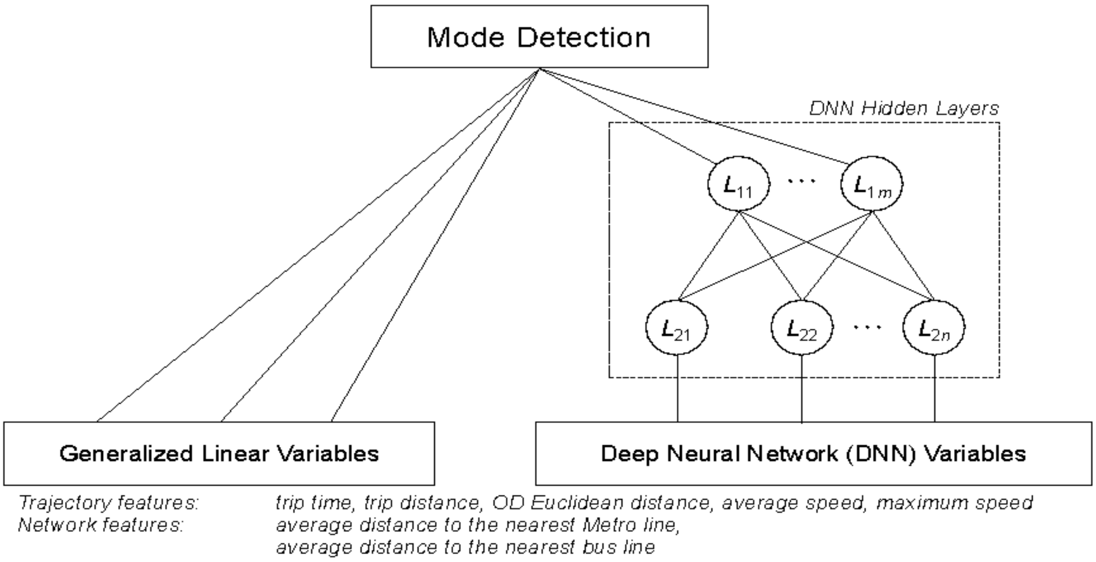

Figure 1. The Framework of the Mode Detection Model Based on Wide and Deep Learning.

3. TRAVEL MODE DETECTION DATA

3.1 Data Collection Effort

The data was originated from a research analyzing the travel behavioral impact of the 2016~2017 Washington D.C.'s Metro SafeTrack project. The SafeTrack is a series of 16 maintenance surges that addresses safety recommendations for the Metro system, which lead to significant service disruptions to different Metro lines in D.C. To assess the impact and analyze the travel behavior responses, the authors have conducted joint web-smartphone surveys on over 2,000 Metro users. A total of 865 trips are specified with travel mode information and these data are used for mode detection modeling in this study. Of these 865 trips, 19.31% are auto trips, 15.84% are bus trips, 52.94% are metro trips and 12.37% trips are walk trips. Since this survey was targeted towards metro users, a high percentage of metro trips are captured. During the survey, only three trips are reported as bike trips and these trips are excluded from this study due to small sample size.

3.2 Data Processing for Training

In this study, the location point data collected in this study has information including latitude, longitude, the instantaneous speed, accuracy and timestamp. The collected raw GPS location data is filtered based on two criteria: accuracy and the average speed between two successive location points. To impute travel mode information, trip end information has to be extracted from a series of GPS location points. The trip end identification method in this study is similar to the approach proposed by Tang, Pan and Zhang (2017). A trip end is identified as the first and last location point in a stay region. In this study, a stay region is defined as the region where the user has stayed longer than a time threshold ⍰, within a distance range of ⍰ and under a speed limit ⍰. The typical trajectory features are employed in our empirical testing as: trip distance, trip time, OD distance, average speed, maximum instantaneous speed, and average data record. In addition, this study also defines network features using the available Metro and bus line information. In specific, we employ the average distance to the nearest rail.

4. EMPIRICAL TESTING RESULTS

4.1 Goodness of Fit Measures

We compare the models discussed in this paper, generalized linear model, decision tree, support vector machine, random forest, adaptive boosting, deep neural network, and wide-and-deep model. The prediction accuracy of 10-fold cross validation is used to measure the performance of the models that we compare. Before that, we checked the performance of the network features. The average accuracy increases from 80.6% to 89.6% in the random forest model and increases from 92.9% to 98.3% in our deep and wide learning model. Therefore, in the next section the network features will be included in both the benchmark and advanced models.

Table 1 summarizes the performance measures of the models. With 400 neuron nodes coded in the first hidden layer and a default optimizer: AdaGrad, the average prediction accuracy of the model reaches 95.7%. Compared to the benchmark models, the improvement is significant. The best Wide and Deep model with RMSProp optimizer can reach 98.3% prediction. We also find that by adding the network features, the detection accuracy rate are greatly enhanced.

Table 1. Goodness of Fit Measures Based on Different Machine Learning Methods

Model	Avg. Loss	Avg. Accuracy
Generalized Linear Model	0.299	0.867
Decision Tree	0.354	0.851
Support Vector Machine	0.238	0.885
Random Forest	0.193	0.896
Adaptive Boosting	0.125	0.908
Deep Neural Network	0.175	0.903
Wide and Deep Model (AdaGrad Optimizer)	0.076	0.957
Wide and Deep Model (RMSProp Optimizer)	0.045	0.983

4.2 Precision and Recall Rates

Besides accuracy, we also need consider the precision and recall rates of our model. The precision (also called positive predictive value) equals to FP/(TP + FP). Here, TP is the number of true positives, and FP is the number of false positives. And recall (also known as sensitivity) equals to TP/(TP + FN), where FN is the number of false negatives. Figure 2.1 and 2.2 in the following compare the benchmark models and the proposed Wide and Deep model (all with network features) by examining the precision and recall rates of the 10-fold cross validation.

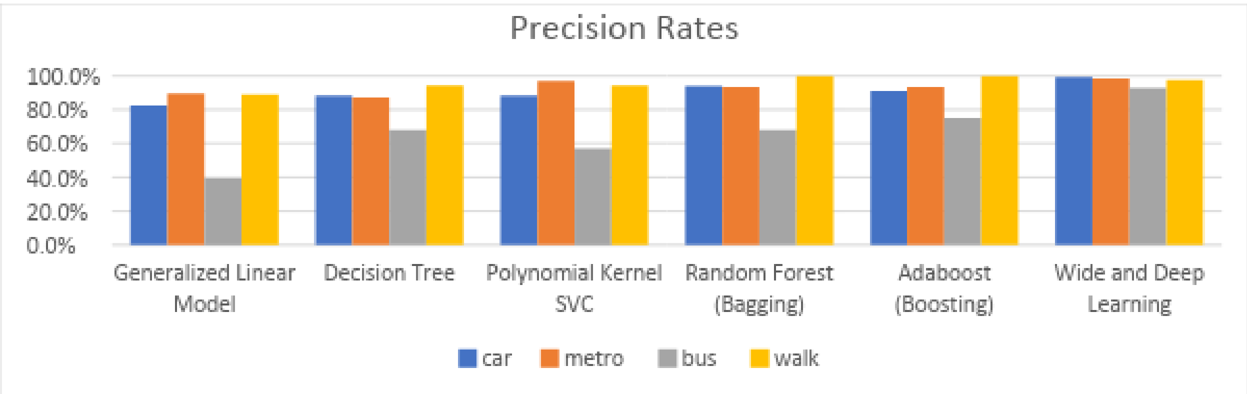

Figure 2.1. Precision Rates of Benchmark Models and the Wide and Deep Learning Model.

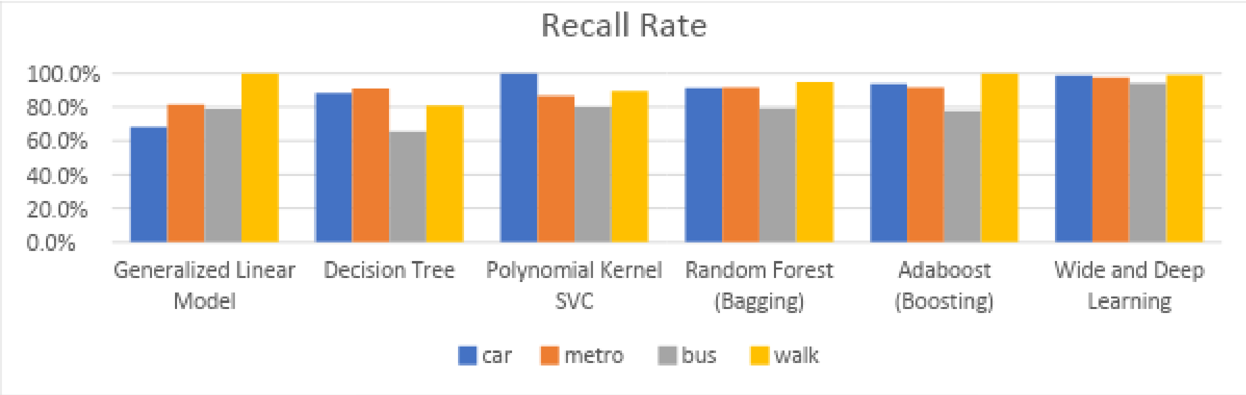

Figure 2.2. Recall Rates of Benchmark Models and the Wide and Deep Learning Model.

Overall, the stand-alone models (Generalized Linear Model, Decision Tree and Support Vector Machine) are worse than the ensemble models (Random Forest and Adaboost). Moreover, the Wide and Deep Learning model has the best results. We could find out from the above two figures that the Wide and Deep Learning Model has the highest average precision and recall rates, especially for metro and bus. The precision and recall rates are all over 90% in the Wide and Deep Learning model, which indicates that each mode is predicted well inside our model. Nevertheless, for other models, the bus mode has both low precision rates and recall rates.

5. CONCLUSION AND DISCUSSIONS

5.1 Conclusion

This paper is a first study that brings a joint wide and deep learning modeling framework to travel mode detection. Being "wide" and "deep" at the same time, this proposed model combines the advantages of both types of models and greatly improves the model goodness-of-fit. The paper also innovatively adopts multimodal network measurements as features to detect Metro and bus modes. The wide and deep learning model has the best prediction results compared to other machine learning models. Follow up research should look into how the model can be fine-tuned even further. With near perfect mode detection accuracy, the replacement of traditional survey method using passively collected data powered by deep learning could be placed on agenda for discussion. For real-time implementations in smart mobility service apps such as the incenTrip, more work on computing is desired in order to bring down the model training time for these wide-and-deep models.

5.2 Discussion on Further Implementations

A trained and validated travel mode detection model enables applications in various domains. Our implementation context is a mobile phone application, incenTrip, which provides real-time and predictive traffic information and incentives to help its users make travel choices. The information and incentives are customized and tailored based on the accurate detection of the app user's actual travel modes, thus making our model crucial to the success of the app. GPS and travel data are collected by the app in the form of raw format, which includes trajectory, speed and time in a pre-defined time interval and fed into the Mode Detection engine. The training of the wide and deep model also takes place in the Mode Detection Engine. Based on the findings reported in the previous sections, the training and updating of the model take less than 5 minutes to reach 98% prediction accuracy. Explorations of the other possible implementation of the model can include research to understand revealed-preference travel mode choice. The study can be potentially applied to evaluate to what extent passively collected GPS survey data can be used to replace expensive household travel and activity diaries. If combined with information and incentive provision technology such as incenTrip, it is possible to learn trivial behavioral insights on human reactions to such interventions.

ACKNOWLEDGEMENT

The research is financially supported by the Maryland State Highway Administration (MD-SHA) and the Department of Energy ARPA-E TRANSNET Program. The opinions in this paper do not necessarily reflect the official views of MD-SHA or U.S. DOE. They assume no liability for the content or use of this paper. The authors are solely responsible for all statements in this paper.

REFERENCES:

Byon, Y.; Liang, S. (2014). Real-time transportation mode detection using smartphones and artificial neural networks: Performance comparisons between smartphones and conventional global positioning system sensors. J. Intell. Transp. Syst. 2014, 18, 264–272.

Cheng, H. T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., ... & Anil, R. (2016). Wide & deep learning for recommender systems. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems (pp. 7-10). ACM.

Freund, Y., Schapire, R., & Abe, N. (1999). A short introduction to boosting. Journal-Japanese Society For Artificial Intelligence, 14(771-780), 1612.

Gonzalez, P.A.; Weinstein, J.S.; Barbeau, S.J.; Labrador, M.A.; Winters, P.L.; Georggi, N.L.; Perez, R. (2010) Automating mode detection for travel behaviour analysis by using global positioning systems-enabled mobile phones and neural networks. IET Intell. Transp. Syst. 2010, 4, 37–49.

Ho, T. K. (1995). Random decision forests. Proceedings of the Third International Conference on Document Analysis and Recognition, 1, 278-282). IEEE.

Lari, Z.A. and Golroo, A. (2015). Automated Transportation Mode Detection Using Smart Phone Applications via Machine Learning: Case Study Mega City of Tehran. In Proceedings of the Transportation Research Board 94th Annual Meeting, Washington, D.C., 11~15 January 2015.

Stenneth, L.; Wolfson, O.; Yu, P.S.; Xu, B. (2011). Transportation Mode Detection Using Mobile Phones and GIS Information. In Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Chicago, IL, USA, 1–4 November 2011.

Tang, L., Pan, Y., and Zhang, L. (2018). Trip purpose imputation based on long-term GPS data. In proceedings of the 2018 Transportation Research Board Annual Meeting. Jan 8-11, 2018, Washington D.C.

Tang, L., Xiong, C., and Zhang, L. (2015). Decision tree method for modeling travel mode switching in a dynamic behavioral process. Transportation Planning and Technology. 38(8), 833-850.

Wu, L., Yang, B., Jing, P. (2016). Travel mode detection based on GPS raw data collected by Smartphones: A systematic review of the existing methodologies. *Information*. 7 (67). doi:10.3390/info7040067

Xiao, G.; Juan, Z.; Gao, J. (2015) Travel Mode Detection Based on Neural Networks and Particle Swarm Optimization. Information 2015, 6, 522–535.

Xiong, C. and Zhang, L. (2013). A descriptive Bayesian approach to modeling and calibrating drivers' en-route diversion behavior. IEEE Transactions on Intelligent Transportation Systems. 14(4). 1817-1824.

Zheng, Y., Wang, L., Liu, L., & Xie, X. (2011). U.S. Patent No. 8,015,144. Washington, DC: U.S. Patent and Trademark Office.