

New Driver Workload Prediction Using Clustering-Aided Approaches

Dewei Yi^{ID}, Jinya Su^{ID}, Cunjia Liu^{ID}, and Wen-Hua Chen^{ID}, *Fellow, IEEE*

Abstract—Awareness of driver workload (DW) plays a paramount role in enhancing driving safety and convenience for intelligent vehicles. The DW prediction systems proposed so far learn either from individual driver’s data (termed personalized system) or existing drivers’ data indiscriminately (termed average system). As a result, they either do not work or lead to a limited performance for new drivers without labeled data. To this end, we develop clustering-aided approaches exploiting group characteristics of the existing drivers’ data. Two clustering aided predictors are proposed. The first is clustering-aided regression (CAR) model, where the regression model for the cluster with the highest likelihood is adopted. The second is clustering-aided multiple model regression model, where the concept of multiple models is further augmented to CAR. A recent dataset from real-world driving experiments is adopted to validate the algorithms. Comparative results against the conventional average system demonstrate that by incorporating clustering information, both the proposed approaches significantly improve workload prediction performance.

Index Terms—Classification and regression tree (CART), clustering, multiple model, workload inference.

I. INTRODUCTION

RECENT years have witnessed an increasing interest in developing advanced driver assistance systems (ADASs) and in-vehicle information systems (IVISs) to enhance driving safety and convenience. ADASs can potentially improve safety by augmenting drivers’ situation awareness accuracy and alerting drivers to potential dangers [1]–[3]. While IVISs bring convenience by offering real-time advice, instructions, and entertainment through navigation system, music, etc. [4]. It was, however, reported that these functions, if not employed appropriately, may increase driver’s distraction and workload [4]. This instead brings adverse effects such as annoying the driver or even increasing the risk of traffic accidents.

Manuscript received July 30, 2017; revised June 20, 2018; accepted September 11, 2018. Date of publication October 9, 2018; date of current version December 14, 2018. This work was supported by the U.K. Engineering and Physical Sciences Research Council Autonomous and Intelligent Systems Programme with BAE Systems as the leading industrial partner under Grant EP/J011525/1. This paper was recommended by Associate Editor X. Xu. (*Corresponding author: Jinya Su.*)

The authors are with the Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough LE11 3TU, U.K. (e-mail: d.yi@lboro.ac.uk; j.su2@lboro.ac.uk; c.liu5@lboro.ac.uk; w.chen@lboro.ac.uk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSMC.2018.2871416

A promising solution is to monitor driver workload (DW) in real time so that these functions can adapt to driver’s workload, i.e., providing “adaptive aiding” [4]–[6]. For instance, an earlier collision warning signal can be provided to a driver under high workload [5], [7]; HMI can also be optimized [8] such as switching off certain functions under high workload. It is prerequisite to accurately monitor DW in real time. Several challenges, however, exist. First, DW is not directly measurable. Second, various signals are available for workload inference, however, it is unclear which feature combination is most effective. Third, labeling DW is challenging, costly, and time-consuming [9].

Due to its significance in enhancing safety and convenience, DW prediction has been drawing increasing attention [10]–[13]. Researchers from different fields have devised various algorithms to build an implicit mapping between diagnostic signals and DW. The existing approaches are categorized into three classes according to the type of measurements in mapping building, including physiological signals [11], vision signals [12], and vehicle-related signals [6], [13]. Please also refer to survey paper [10] for more information. The DW prediction system (DWPS) can also be categorized into “average systems” and “personalized systems” according to different datasets for model construction. In average systems, an average model is learned from existing drivers’ data indiscriminately. While in personalized systems, a personalized model dedicated to individual drivers is learned from individuals’ historical data. The personalized approach has been proved to be more effective than an average model since personalized driving characteristics have been accommodated [6].

In practical applications when a new driver starts to adopt a DWPS, no historical data is available for the new driver. So a personalized system is not applicable. One has to rely on an average system, which may result in a limited performance since the driving characteristics have not been accommodated [14]. Therefore, this paper aims to tackle the problem of workload prediction for new drivers so that user confidence in DWPSs is not compromised due to the new user problem.

To maximally exploit existing drivers’ data, we develop clustering-aided predictors exploiting group characteristics of existing drivers. Fuzzy *C*-means (FCM) clustering is first adopted to cluster existing drivers’ data. Then classification and regression tree (CART) is drawn to learn classification tree model (CTM) and regression tree model (RTM), where data within a cluster is to train RTM and the clustered data is to train CTM. Two clustering-aided algorithms are developed. The first one is clustering-aided regression (CAR), where one

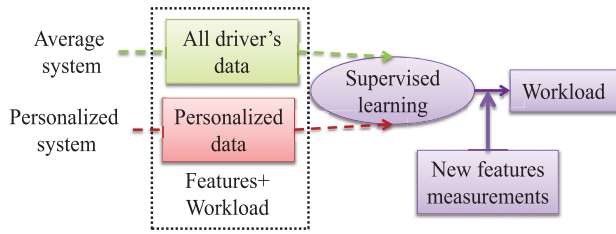


Fig. 1. Conventional workload prediction systems.

RTM model is adopted for workload prediction using the cluster with the highest probability from CTM. To attenuate the effect of classification error, the concept of “multiple models” [15] is also applied by treating the likelihood from CTM as weighting for each RTM. This leads to clustering-aided multiple model regression (CAMMR).

A recently collected dataset from real-world driving experiments in 2013 [9] is adopted for algorithm validation. Different approaches including conventional average regression without clustering (ARWOC) model and the proposed CAR and CAMMR models are compared. Comparative results demonstrate that by incorporating clustering information or multiple models, the proposed CAR and CAMMR substantially improve workload prediction performance. The idea of clustering-aided approach or multiple models is not new, and has been applied to position prediction and regression analysis in [2] and [16]. However, to the best of our knowledge, these techniques have not been exploited for workload prediction and their integration is novel. More precisely, the contributions of this paper are summarized as follows.

- 1) Several regression algorithms are compared to identify a suitable one for DW prediction, where CART with Bayesian optimization outperforms others.
- 2) For the first time, the concepts of CAR and CAMMR are introduced to the problem of new DW prediction accommodating group characteristics.
- 3) A dataset from real-world driving experiments is to validate the proposed algorithms with promising results.

II. NEW DRIVER WORKLOAD PREDICTION

As discussed in the introduction, the existing workload prediction systems are defined in Definitions 1 and 2, where their diagrams are depicted in Fig. 1.

Definition 1 (Average System): In this system, an average workload predictive model (or a model for all drivers) is learned from all available drivers’ data indiscriminately. Then this model is directly applied to new drivers.

Definition 2 (Personalized System): In this system, a personalized predictive model dedicated to individual drivers is learned from individual drivers’ data rather than all drivers’ data.

A. Research Motivations

Different from conventional studies for average workload model construction [10], this paper considers new DW prediction with the following definition.

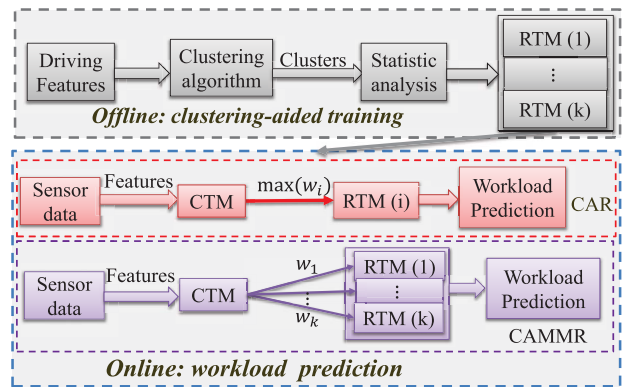


Fig. 2. Clustering-aided predictors for new DW prediction: offline (upper plot); online (lower plot); red blocks for CAR; and purple blocks for CAMMR.

Definition 3 (New User Problem): In this paper, denotes the scenario where a new driver starts to adopt a DWPS and no labeled data regarding the new driver is available.

It can be seen that for new user problem, the average system can work but may lead to a degraded performance, since the characteristics of individual drivers have not been effectively accommodated [6]. While compared to average system the personalized system is more accurate when sufficient labeled individual data is available [14]. Unfortunately, labeling DW is challenging, costly, and time-consuming [9]. Therefore, with a limited or sparse dataset, it is difficult if not possible to learn an accurate relationship and in turn provide a poor workload prediction. To this end, clustering-aid approaches are proposed to enhance workload prediction performance. Our main idea is to cluster the existing drivers’ data into different groups and identify a suitable group (or a weighting for each group) for the new driver.

B. Clustering-Aided Approaches

Our clustering-aided approaches comprise offline training and online execution. In offline training, clustering algorithms are adopted to cluster the existing drivers’ data. Then CART is further drawn to explore the data, resulting in CTM and RTM. The data within a cluster is to train RTM, while the clustered data is to train CTM. More specifically, RTM is to mapping features with workload, and CTM is to classify samples into different clusters with corresponding probabilities.

In online execution, features are first fed into CTM so that the probability for each cluster is returned. Then two clustering-aided predictors are developed. The first is CAR, where only one RTM model corresponding to the cluster with the highest probability is adopted. The concept of multiple models is also augmented to CAR by treating the likelihood from CTM as the weighting for each RTM, leading to CAMMR. The overall diagram of the proposed clustering-aided predictors is depicted in Fig. 2.

Theoretical properties of the proposed methods are also discussed. Both CAR and CAMMR rely on clustering and classification to accommodate the group characteristics. In particular, CAR only utilizes one regression model dedicated to the cluster with the highest probability. While CAMMR, using

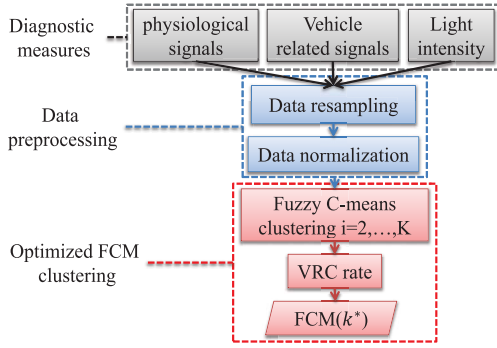


Fig. 3. Flowchart for clustering analysis.

the idea of multiple model [17], runs a number of regression algorithms simultaneously with different weightings for each regression model. Consequently, CAR outperforms average model due to the introduction of group characteristics via clustering analysis; while CAMMR can further improve the robustness of CAR by accounting for the possible classification error. It is also noted that the developed systems are different from semi-supervised learning methods [18], [19]. This is because in semi-supervised learning methods unlabeled data is adopted to either modify or reprioritize hypotheses obtained from labeled data alone [18], while the developed systems are focused on how to maximally exploit the labeled data (i.e., existing drivers' data) for new driver (i.e., unlabeled data) workload prediction.

III. CLUSTERING ANALYSIS

Clustering analysis is considered in this section with overall framework in Fig. 3. Some key elements of the framework are first elaborated in the following sections.

A. Diagnostic Signals

In this paper, several types of measurements available in the dataset [9] are pooled together as diagnostic signals. Specifically, they include driver-related physiological signals, [e.g., electrocardiogram (ECG), skin conductance response (SCR), body temperature (BT), heart rate (HR), and heart rate variability (HRV)], vehicle-related signals (e.g., GPS and IMU measurements) and driving environment-related signal (e.g., intensity of light). These features have been proved to be able to effectively reflect DW [10].

B. Data Preprocessing

Features are usually measured in different frequencies and with different ranges. So data preprocessing is usually involved to achieve a better performance. Two preprocessing steps are involved in this paper, including data resampling and normalization, where resampling is to unify features' length and normalization is to improve algorithm efficacy. Z-score [20] is adopted for data normalization, which can retain shape properties of the original data such as skewness and kurtosis. For a vector x , Z-score is defined by $z = (x - \bar{x})/\sigma$ with \bar{x} and σ being mean and standard deviation (Std).

Algorithm 1: FCM Clustering With Optimized Cluster Number

1. Given a user-defined maximum cluster number K ;
2. Perform FCM clustering under cluster number $i = 2, \dots, K$ resulting in $FCM(2), \dots, FCM(K)$;
3. Calculate VRC rate for each clustering using formula (1) resulting in $VRC(2), \dots, VRC(K)$;
4. $FCM(k^*)$ is the optimal clustering configuration, with optimal cluster number $k^* = \arg \max_{i=2}^K VRC(i)$.

C. FCM Clustering With Optimized Cluster Number

Then FCM clustering is adopted for clustering analysis due to its fine properties including a better performance for overlapped data [2] by incorporating uncertain information [21]. Its detailed derivation is omitted due to a lack of space, interested readers may refer to [6, Sec. III-B] for more information. It is generally not easy to determine the cluster number for FCM clustering, which in this paper is optimized by using Calinski–Harabasz criterion, also termed variance ratio criterion (VRC),

$$VRC(k) = \frac{SS_B}{SS_W} \times \frac{(n-k)}{(k-1)} \quad (1)$$

where SS_B and SS_W denote overall between-cluster and within-cluster variance, and k denotes cluster number. SS_B is defined by $SS_B = \sum_{i=1}^k n_i \|c_i - m\|^2$, where c_i and n_i denote the centroid and the number of data points in the i th cluster, m is the overall mean of the sample data, and $\|\bullet\|^2$ denotes Euclidean distance. SS_W is defined by $SS_W = \sum_{i=1}^k \sum_{x \in C_i} \|x - c_i\|^2$, where x is a data point, C_i is the i th cluster. It is shown in [22] that a larger VRC value means a better data partition. So determination of the “optimal” cluster number reduces to maximizing $VRC(k)$ with respect to k . The overall clustering algorithm is summarized in Algorithm 1.

IV. CLUSTERING-AIDED PREDICTORS

Clustering-aided predictors for new DW prediction are detailed in this section. Before that some key elements are first introduced.

A. Regression Model Selection

Our framework relies on regression analysis to build a mapping between diagnostic signals and workload. It is not easy to select an appropriate algorithm. Therefore, different regression models are first compared using existing drivers' data. The algorithms contain six different categories, including linear regression [23], CART [24], support vector regression (SVR) [25], Gaussian process regression (GPR) [26], ensemble learning [27], and (deep) neural network [28] and their variants. According to the results in Section V-G, CART with complex structure outperforms others and so is adopted. The CART model is briefly introduced in the following section.

B. CART Model

In CART model, the classification and regression model is trained by building a binary decision tree according to certain

Algorithm 2: CART Model

1. Given a sequence of examples S and a set of discrete attributes A ;
 2. For each attribute $a^i \in A$, the set of attribute values A^i is partitioned into two disjoint subsets A_L^i and A_R^i , let X^i denote all possible A_L^i ;
 3. For each $A_L^i \in X^i$, calculate the Gini gain/Mean Squared Error (MSE) and select the variable ($\tilde{A}_{L,q}^i = A_L^i$) which maximizes Gini gain or minimizes MSE ($\tilde{A}_{L,q}^i$ is a split);
 4. Send data $S(\tilde{A}_{L,q}^i)$ to the “leaf node” and $S(A_{R,q}^i)$ to the “right node”;
 5. Recursively repeat the same process on these two “nodes” until stopping rules are satisfied;
-

splitting rules (for node determination and threshold selection) based on feature variables. The detailed algorithm is referred to [29], where the pseudocodes are summarized in Algorithm 2 for the sake of completeness.

Remark 1: There are generally certain stopping rules in CART to avoid its adverse effects such as overfitting. In this paper, one of the stopping rules is considered, i.e., the minimum leaf size. A smaller leaf size may make the model more prone to capturing noise in training data. The determination of minimum leaf size is transformed into the minimization of the cross validation (CV) loss for CART. And the optimality problem is solved by using Bayesian optimization [30].

C. Clustering-Aided Predictors

As depicted in Fig. 2, two clustering-aided algorithms are proposed in this paper, including CAR and CAMMR, which are detailed as below.

1) *CAR Algorithm:* As depicted in Fig. 2, given a new sample, its cluster information can be first determined by CTM. The predicted class with the highest probability from CTM takes the following form:

$$\hat{y} = \arg_{y=1,\dots,k^*} \min \sum_{i=1}^{k^*} \hat{P}(i|x) C(y|i) \quad (2)$$

where k^* is class number, $\hat{P}(i|x)$ is the posterior probability of class i given observation x , and $C(y|i)$ is the cost of misclassification, where the default form is $C(y|i) = 1$ for $y \neq i$ and $C(y|i) = 0$ for $y = i$. Suppose the predicted class is \hat{y} for a given sample, one can derive CAR algorithm, where one RTM for cluster \hat{y} , $\text{RTM}(\hat{y})$, is used for workload prediction. The DW of CAR algorithm is determined by the following regression model:

$$\text{DW} \sim \text{RTM}(\hat{y}). \quad (3)$$

2) *CAMMR Algorithm:* To attenuate the adverse effects of classification error, we further apply the concept of multiple models. In this approach, a number of predictive models are run in parallel and the ultimate predicted value is a weighting of each predictive model.

It follows from (2) that the posterior probability $\hat{P}(i|x)$ for each class i is also generated, which can be used

to calculate the weighting for each predictor. Suppose $\text{RTM}(1), \dots, \text{RTM}(k^*)$ are k^* regression models corresponding to different clusters, and w_1, \dots, w_{k^*} are their weighting derived from CTM algorithm using the formula

$$w_i = \frac{\hat{P}(i|x)}{\sum_{i=1}^{k^*} \hat{P}(i|x)}, \quad i = 1, \dots, k^*. \quad (4)$$

One can derive CAMMR, where the DW of CAMMR algorithm is determined by the following regression model:

$$\text{DW} \sim \sum_{i=1}^{k^*} \text{RTM}(i) w_i. \quad (5)$$

Remark 2: Only one RTM is adopted in CAR and a parallel of RTMs with corresponding weighting w_i are adopted in CAMMR. So CAMMR is more robust against the classification error of CTM, however, this is at the price of a higher computational cost. This observation will be demonstrated in Section V-G.

V. EXPERIMENTAL VALIDATION

In this section, experimental validation is considered, where all algorithms are implemented in MATLAB 2017a under Windows 7 Operation System and are evaluated on a PC with the following configuration: Intel Core i5-CPU at 3.20 GHz with 16 GB of RAM.

A. Driving Dataset

The dataset for algorithm validation is from real-world driving experiments of about 30 min with ten participants of various background, which was collected by Human-Computer Interaction (HCI) Laboratory, University of Stuttgart in 2013 [9]. In the experiment, driver-related physiological signals (i.e., skin conductance response, heart rate, skin temperature, and ECG recorded at 128 Hz) and vehicle-related measurements (i.e., GPS position and speed recorded at 1 Hz, and acceleration recorded at 12 Hz) are measured along with environment-related signals (i.e., light intensity) via cameras.

Two webcams were available to record the driving scenario and the driver at 29 frames/s. After the driving test, each participant was required to perform a *post-hoc* video rating evaluating the perceived workload in the range of 0 (no workload) and 1000 (maximum workload). This method is also termed subjective rating approach [12], which is a common approach to deriving workload ground truth data. Data extrapolation has also been done by Harbluk *et al.* [12] to create a uniformed dataset at 128 Hz.

To summarize, the model inputs are composed of physiological signals (i.e., ECG, SCR, BT, HR, and HRV), vehicle-related signals (i.e., speed of GPS, GPS latitude, GPS longitude, GPS altitude, longitudinal acceleration, lateral acceleration, vertical acceleration, GPS bearing, and GPS accuracy), and light intensity. The model output is workload from subjective rating by each driver. Therefore, the dimension of input and output data are n -by-15 and n -by-1 with n being sample number. After data synchronizing, the total data size for all ten drivers is 2 298 870 with a sampling rate

TABLE I
VRC OVER CLUSTER NUMBER IN VARIOUS CVS

CV/k	2	3	4	5	6
1	3461.99	3210.79	1749.57	2656.97	2502.63
2	1987.15	3061.90	2815.12	1746.16	2619.90
3	1952.50	1799.89	1707.63	2597.70	2460.06
4	3451.80	3078.97	2794.79	2625.02	2428.91
5	1977.92	3059.90	2692.84	2622.46	2573.48
6	3477.99	3333.65	1854.09	2776.54	2484.21
7	2258.43	3239.77	2903.09	2613.47	2561.06
8	2035.29	3133.23	2862.63	2367.85	2545.92
9	2096.70	3137.86	2638.35	2654.33	2437.09
10	3444.06	3146.41	1725.80	2626.06	2444.32

of 128 Hz. More details about the dataset (e.g., format) are referred to [9]; the dataset can also be freely accessed via www.hcilab.org/research/hcilab-driving-dataset.

B. *k*-Fold Cross Validation

To effectively exploit the limited driving data, *k*-fold CV [31] is adopted. Since we have driving data of ten participants, tenfold CV is used. More precisely, in each CV one out of ten participants is selected as the new driver and the reminding nine participants are assumed to be existing drivers whose data are used for model training.

C. FCM Clustering With Optimal Cluster Number

FCM clustering with optimal cluster number is first experienced. The fuzzy overlap controller is chosen $m = 2$ by following [21]. Since tenfold CV is adopted, the training/testing sets and consequently division of clusters for existing drivers' data differ in each time. The VRC values under different cluster numbers in different CVs are summarized in Table I. It can be seen that the optimal cluster numbers for different CVs are 2, 3, 5, 2, 3, 2, 3, 3, 3, and 2, respectively.

D. Regression Algorithm Selection

Regression algorithm selection is further considered, where the candidate algorithms include linear regression, CART decision tree, SVR, GPR, ensemble learning, (deep) neural network, and their variants. Particularly, simple tree, medium tree, and complex tree are variants of CART, where the maximum number of splits are 4, 20, and 100, respectively.

To evaluate their performance, the widely used metrics including mean absolute error (MAE) and *R*-squared are adopted, where MAE measures prediction accuracy and *R*-squared evaluates data fitting quality. Moreover, both training time and prediction speed are compared. The comparative results are summarized in Table II.

It follows from Table II that: 1) CART with complex tree outperforms others in terms of MAE and *R*-squared; 2) training time of CART is very short, only slightly longer than linear regression; and 3) prediction speed of CART is the fastest among the four types of regression models. Considering the prediction performance along with training and prediction time, CART model with complex tree structure is adopted for subsequent algorithm/system design.

TABLE II
PERFORMANCE OF DIFFERENT REGRESSION ALGORITHMS

Classifier	R-squared	MAE	Prediction (obs/sec)	Training (sec)
Linear Regression				
Linear	0.31	148.45	110000	29.404
Interactions Linear	0.46	127.99	25000	38.047
Stepwise Linear	0.45	128.63	46000	19698
CART				
Complex Tree	0.93	23.14	120000	60.908
Medium Tree	0.90	33.00	93000	59.136
Simple Tree	0.85	49.93	150000	55.261
SVR				
Quadratic	0.48	118.70	1400	1566.4
Cubic	0.52	93.07	1500	5071.5
Fine Gaussian	0.78	70.05	940	2600
GPR				
Squared Exponential	0.91	42.30	810	18831
Matern 5/2	0.92	38.13	600	25992
Rational Quadratic	0.92	37.61	480	50869
Ensemble Learning				
Random Forest	0.95	29.89	10000	2962.4
Adaboost	0.65	101.25	30000	2758.7
(Deep) NN				
Back Propagation	0.90	75.75	8938	378.12
Multilayer Perceptron	0.79	107.88	71	1205.1

E. Feature Reduction and Selection

Feature reduction can generate a simple (i.e., with fewer model parameters) but effective (e.g., high accuracy and good generality) model, especially for the scenario where only a limited number of labeled samples are available [32]. Various dimension reduction algorithms can achieve this task such as feature selection and feature transformation [33]. In this paper, only feature selection is considered, since feature transformation will make features' physical meanings lost. Different types of feature selection algorithms are available in the literature such as filter methods, wrapper methods, and embedded methods [32]. In this paper, filter method is adopted due to its simplicity, efficiency, and independence of the regression algorithms.

In particular, the classical minimum redundancy and maximum relevance (MRMR) in [34] is adopted, which considers the relevance between features and label, and feature redundancy concurrently. Each feature is first ranked by MRMR and the top features are selected for prediction model construction. Feature ranking and prediction performance under different numbers of top features are displayed in Table III. It follows from Table III that the prediction model by using all features results in the best performance. The possible reasons are: 1) feature dimension is not very high; 2) each feature has certain usefulness in predicting DW; and 3) different features reflect different aspects of workload and little feature redundancy exists. Therefore, all 15 features are still adopted in the following results.

F. Optimized CART Model

After CART is chosen as the regression model in Section V-D, it is further optimized by Bayesian parameter optimization. In this paper, the maximum number of iterations is chosen 30, the value of minimum leaf size is searched

TABLE III
MRMR FEATURE RANKING AND PERFORMANCE UNDER DIFFERENT
SETS OF TOP FEATURES

Feature	Ranking	Selected Feature(s)	MAE
ECG	1	Top 1 Ranked	172.66
Longitude	2	Top 2 Ranked	87.29
Intensity of Light	3	Top 3 Ranked	58.74
Bearing	4	Top 4 Ranked	49.71
SCR	5	Top 5 Ranked	29.61
Altitude	6	Top 6 Ranked	28.68
HRV	7	Top 7 Ranked	27.40
Speed	8	Top 8 Ranked	26.47
HR	9	Top 9 Ranked	30.08
BT	10	Top 10 Ranked	25.43
GPS Accuracy	11	Top 11 Ranked	25.46
Lateral Acceleration	12	Top 12 Ranked	27.13
Longitudinal Acceleration	13	Top 13 Ranked	26.72
Vertical Acceleration	14	Top 14 Ranked	26.91
Latitude	15	All Features	23.14

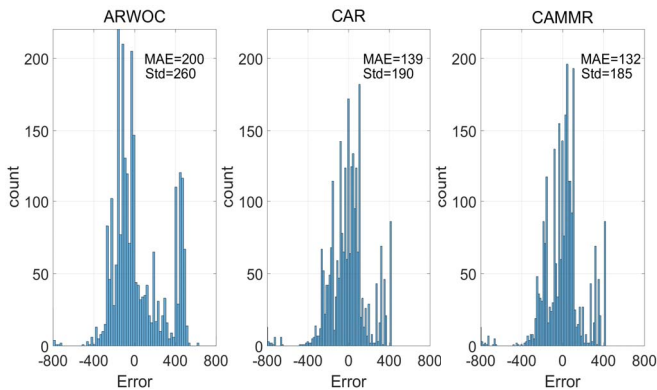


Fig. 4. Workload prediction error histogram for participant 2 under ARWOC, CAR, and CAMMR.

among log-scaled in the range of $[1, \max(2, n - 1)]$ with n being the number of observations. The depth of the tree can be controlled by maximal number of decision splits, min leaf size (MLS), and min parent size (MPS). In this paper, MPS is determined by $MPS = \max[M, 2 * MLS]$, with M being a fixed number of 10. The exemplary results are omitted due to a lack of space and are available upon request.

G. Average System Versus Clustering-Aided Systems

In this section, the proposed CAR and CAMMR are experimented against a baseline average system using all existing drivers' data indiscriminately (termed ARWOC). In the average ARWOC system, one driver is treated as the new driver and the remaining nine drivers are treated as the existing drivers, where the optimized CART algorithm is directly learned by using the nine existing drivers' data indiscriminately.

Without loss of generality, the workload prediction error histogram for participant 2 under three different approaches are depicted in Fig. 4 as an illustrating example. Along with prediction error histogram, two classical metrics for regression analysis are also calculated, including MAE and Std, where MAE assesses the average prediction performance and Std assesses the stability of the algorithm.

TABLE IV
MAE COMPARISONS FOR DIFFERENT APPROACHES

No.	ARWOC	CAR	CAMMR
1	406.02	312.46	312.14
2	199.75	133.63	132.47
3	223.56	196.11	187.88
4	214.00	169.01	165.17
5	265.15	408.20	408.38
6	316.86	228.35	224.12
7	279.52	226.49	225.28
8	262.24	246.55	245.55
9	202.27	167.06	163.89
10	216.85	227.36	226.36
Average	258.62	231.52	229.12
Prediction Speed	1136	1210	614

It follows from Fig. 4 that CAR and CAMMR substantially outperform conventional ARWOC without clustering information in terms of accuracy (MAE) and stability (Std). For example, the MAE/Std of CAR (139/190) and CAMMR (132/185) are much smaller than that of ARWOC (200/260). The comparative statistical results in term of MAE for all participants and the average prediction speed (observations per second) are summarized in Table IV.

The following observations can be drawn from Table IV.

- 1) Except drivers 5 and 10, CAR and CAMMR significantly outperform the ARWOC, which demonstrates that by incorporating group characteristics of training data, workload prediction performance can be improved.
- 2) CAMMR using the concept of multiple models outperforms CAR in term of prediction accuracy.
- 3) CAR and ARWOC have a similar computation time, while CAMMR takes a longer time than the other two approaches, but is still valid for real time applications.

To summarize, the proposed clustering-aided algorithms including CAR and CAMMR significantly outperform the conventional average system ARWOC in term of prediction accuracy. While CAMMR obtains a better performance than CAR algorithm, however, this is at the price of a slightly higher computation load. Consequently, users can make a choice between the proposed CAR and CAMMR systems depending on the priority of accuracy or computation cost according to different application scenarios.

VI. CONCLUSION AND DISCUSSION

Precise workload prediction for new drivers is challenging due to its nonmeasurability and a lack of personalized data. This paper addressed the issue by maximally exploiting the existing drivers' data. Specifically, clustering is first employed to cluster the existing drivers' data. Then the similarity between a new driver and existing driver cluster is exploited by using classification. Two clustering-aided predictors are proposed including CAR with one suitable regression model and CAMMR where a series of regression models for each cluster are run in parallel with corresponding weightings.

The proposed clustering-aided approaches are validated by a recent dataset from real-world driving experiments. Comparative experimental results against a baseline average

system demonstrate that by incorporating clustering information, the workload prediction performance can be significantly improved. Besides, the concept of multiple models can mitigate the effect of classification error, but inevitably at the price of a slightly higher computation cost.

This paper mainly focuses on proposing a new framework for new DW prediction considering the limitations of existing approaches in handling new user problem. The feasibility of the framework is initially demonstrated using a small real-world driving dataset. It is believed that with more labeled drivers' data, particularly with diverse background, and more advanced learning algorithms, the performance of the proposed approaches can be substantially improved. The following aspects are identified for further development.

- 1) Label information can be incorporated into FCM so that better clustering results can be obtained.
- 2) Incremental learning can be possibly drawn to adopt to new input data over time.
- 3) Semi-supervised learning can be possibly drawn to reduce the burden of ground truth data labeling.

ACKNOWLEDGMENT

D. Yi would like to thank Chinese Scholarship Council for supporting his study in the U.K.

REFERENCES

- [1] J. Zhang *et al.*, "Data-driven intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1624–1639, Dec. 2011.
- [2] D. Yi, J. Su, C. Liu, and W.-H. Chen, "Data-driven situation awareness algorithm for vehicle lane change," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, 2016, pp. 998–1003.
- [3] J. Su and W.-H. Chen, "Model-based fault diagnosis system verification using reachability analysis," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: [10.1109/TSMC.2017.2710132](https://doi.org/10.1109/TSMC.2017.2710132).
- [4] J. Ziegler, T. Hussein, D. Münter, J. Hofmann, and T. Linder, "Generating route instructions with varying levels of detail," in *Proc. 3rd Int. Conf. Autom. User Interfaces Interact. Veh. Appl.*, 2011, pp. 31–38.
- [5] W. Hajek, I. Gaponova, K. H. Fleischer, and J. Krems, "Workload-adaptive cruise control—A new generation of advanced driver assistance systems," *Transp. Res. F Traffic Psychol. Behav.*, vol. 20, pp. 108–120, Sep. 2013.
- [6] D. Yi, J. Su, C. Liu, and W.-H. Chen, "Personalized driver workload inference by learning from vehicle related measurements," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: [10.1109/TSMC.2017.2764263](https://doi.org/10.1109/TSMC.2017.2764263).
- [7] T. Liu, Y. Yang, G.-B. Huang, Y. K. Yeo, and Z. Lin, "Driver distraction detection using semi-supervised machine learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1108–1120, Apr. 2016.
- [8] C. J. D. Patten, A. Kircher, J. Östlund, L. Nilsson, and O. Svenson, "Driver experience and cognitive workload in different traffic environments," *Accid. Anal. Prevent.*, vol. 38, no. 5, pp. 887–894, Sep. 2006.
- [9] S. Schneegass, B. Pfleging, N. Broy, F. Heinrich, and A. Schmidt, "A data set of real world driving to assess driver workload," in *Proc. 5th Int. Conf. Autom. User Interfaces Interact. Veh. Appl.*, 2013, pp. 150–157.
- [10] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt, "Driver behavior analysis for safe driving: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 3017–3032, Dec. 2015.
- [11] J. A. Healey and R. W. Picard, "Detecting stress during real-world driving tasks using physiological sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 156–166, Jun. 2005.
- [12] J. L. Harbluk, Y. I. Noy, P. L. Trbovich, and M. Eizenman, "An on-road assessment of cognitive distraction: Impacts on drivers' visual behavior and braking performance," *Accid. Anal. Prevent.*, vol. 39, no. 2, pp. 372–379, 2007.
- [13] G. A. Meiring and H. C. Myburgh, "A review of intelligent driving style analysis systems and related artificial intelligence algorithms," *Sensors*, vol. 15, no. 12, pp. 30653–30682, 2015.
- [14] S. Schnelle, J. Wang, H. Su, and R. Jagacinski, "A driver steering model with personalized desired path generation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 1, pp. 111–120, Jan. 2017.
- [15] K. George and P. Mutalik, "A multiple model approach to time-series prediction using an online sequential learning algorithm," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: [10.1109/TSMC.2017.2712184](https://doi.org/10.1109/TSMC.2017.2712184).
- [16] J. Jeong, M. Leconte, and A. Proutiere, "Cluster-aided mobility predictions," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun. INFOCOM*, 2016, pp. 1–9.
- [17] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: A survey," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 1, pp. 103–123, Jan. 1998.
- [18] X. Zhu, "Semi-supervised learning literature survey," *Comput. Sci., Univ. Wisconsin, Madison, WI, USA, Rep. 1530*, 2005.
- [19] J. Wang, D. Yang, W. Jiang, and J. Zhou, "Semisupervised incremental support vector machine learning based on neighborhood kernel estimation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 10, pp. 2677–2687, Oct. 2017.
- [20] E. Kreyszig, *Advanced Engineering Mathematics*. New York, NY, USA: Wiley, 2010.
- [21] N. R. Pal and J. C. Bezdek, "On cluster validity for the fuzzy C-means model," *IEEE Trans. Fuzzy Syst.*, vol. 3, no. 3, pp. 370–379, Aug. 1995.
- [22] T. Caliński and J. Harabasz, "A dendrite method for cluster analysis," *Commun. Statist. Theory Methods*, vol. 3, no. 1, pp. 1–27, 1974.
- [23] D. A. Freedman, *Statistical Models: Theory and Practice*. Cambridge, U.K.: Cambridge Univ. Press, 2009.
- [24] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and Regression Trees*. New York, NY, USA: CRC Press, 1984.
- [25] C.-H. Wu, J.-M. Ho, and D. T. Lee, "Travel-time prediction with support vector regression," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 276–281, Dec. 2004.
- [26] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*, vol. 1. Cambridge, U.K.: MIT Press, 2006.
- [27] M. Li and Z.-H. Zhou, "Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 6, pp. 1088–1098, Nov. 2007.
- [28] P. Dey, K. Nag, T. Pal, and N. R. Pal, "Regularizing multilayer perceptron for robustness," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 8, pp. 1255–1266, Aug. 2018, doi: [10.1109/TSMC.2017.2664143](https://doi.org/10.1109/TSMC.2017.2664143).
- [29] L. Rutkowski, M. Jaworski, L. Pietruczuk, and P. Duda, "The CART decision tree for mining data streams," *Inf. Sci.*, vol. 266, pp. 1–15, May 2014.
- [30] A. D. Bull, "Convergence rates of efficient global optimization algorithms," *J. Mach. Learn. Res.*, vol. 12, pp. 2879–2904, Jan. 2011.
- [31] I. H. Witten and E. Frank, *Practical Machine Learning Tools and Techniques With Java Implementations, Data Mining*. Los Altos, CA, USA: Morgan Kaufmann, 2000.
- [32] G. Roffo, S. Melzi, U. Castellani, and A. Vinciarelli, "Infinite latent feature selection: A probabilistic latent graph-based ranking approach," in *Proc. Int. Conf. Comput. Vis.*, Jul. 2017, pp. 1407–1415.
- [33] J. Su, D. Yi, C. Liu, L. Guo, and W.-H. Chen, "Dimension reduction aided hyperspectral image classification with a small-sized training dataset: Experimental comparisons," *Sensors*, vol. 17, no. 12, p. 2726, 2017.
- [34] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.