

Short-Term Prediction of Level of Service in Highways Based on Bluetooth Identification

Mark Richard Wilby, Ana Belén Rodríguez González, Rubén Fernández Pozo, and Juan José Vinagre Díaz

Abstract—A precise knowledge about future traffic will eventually open a new era in traffic management. Research has focused on the still unresolved problem of predicting travel time (TT). However, practitioners favor the level of service (LOS) as a meaningful metric that avoids the continuous fluctuations and link-specificity of TT. Evolving from TT to LOS opens a new research line in the field, moving the underlying mathematical problem from *regression* to *classification*. This study proposes a short-term LOS classifier to fulfill this requirement. Given that traffic conditions are mostly free-flow throughout the day, LOS classes are unbalanced. Therefore, we based our predictor on a Random Undersampling Boost algorithm (RUSBoost), especially suited to overcome this issue. We trained and validated this LOS predictor with 12 months of arrival travel time data, captured by a Bluetooth network with 6 links, in real operation on the SE-30 highway (Seville, Spain). This classifier achieved an average recall of 82.8% for prediction horizons up to 15 minutes, reaching 92.5% predicting congestion. We reached this performance by exploiting two facts that we empirically demonstrated: (i) information from every link (even those in the opposite direction) contributes to increase the accuracy of the prediction; and (ii) traffic presents different behavior depending on the day of the week, which we used to segment the data and construct specific classifiers. These promising results show the potential of the proposed LOS predictor, providing a new perspective into traffic forecast and the subsequent traffic management that yields with what practitioners demand.

Index Terms—Level of service, traffic prediction, Bluetooth identification, classifier, travel time.

I. INTRODUCTION

KNOWLEDGE about future traffic states is of an enormous value. It allows drivers to choose the route that minimizes the time invested in the journey and road managers to trigger actuations that would maximize the performance of their transport networks. Both perspectives share the same objective: reducing congestion.

But *congestion* is not a simple physical phenomenon that we can directly measure. *Where* does a congestion precisely start? How *big* is it? *When* does it really end? It is hard to find a unique and common answer to these questions. Consequently

we are forced to find quantitative variables that can indirectly represent congestion and predict their future values.

The scientific community has invested huge efforts in solving this problem during the past four decades [1]. Researchers have proposed a wide variety of approaches using techniques over different variables provided by yet another varied group of sensors. Originally, inductive loops generated traffic variables such as speed, flow, and volume upon which congestion could be characterized. The intrinsic point-like nature of these measurements complicates the problem given that congestions typically spread throughout an area. This is the reason why the work in this field had to shift to other variables like travel time (TT), associated with stretches [2]. This evolution was viable due to the progressive accessibility to floating car data (FCD) and the deployment of automatic vehicle identification (AVI) systems, among which Bluetooth is a cost-effective technology to provide accurate TT estimations [3].

TT prediction is an extremely complex task for two main reasons. First, we can only calculate TT once vehicles have reached the final end of the stretch, thus it is an inherently delayed variable. Second, TT is a statistical measurement that represents the overall behavior of a set of vehicles that started the journey on the same time interval. This leads to wide confidence intervals that are even more evident during the formation and disappearance of a congestion.

On the plus side, traffic managers do not actually require such a precise information like TT as the basis of their knowledge about the future state of the road. What they really seek for is anticipating significant changes on traffic behavior that they have to respond to. In this respect, the level of service (LOS) [4] arises as an appropriate decision support tool [5]. Unexpectedly, the LOS has not ever been used as the target variable for traffic forecasting.

Setting the objective of prediction on the LOS drastically changes the problem statement and the techniques to solve it. In the first instance, it makes the problem evolve from *regression* to *classification*, which alleviates the complexity introduced by TT. In addition, it allows a better generalization of the results as it avoids the dependency of TT on the length of the stretch under study. LOS is a normalized variable thus directly transferable to any type of stretch and road configuration.

Considering this motivation, the current paper proposes a short-term LOS predictor based on data produced by AVI systems to forecast future traffic conditions. These data reflect that traffic does not often fall on the category of heavy

congestion, which produces LOS information that is clearly unbalanced. In order to deal with this fact, we developed a Random Undersampling Boost (RUSBoost) classifier able to optimally adapt to this situation. This classifier cannot be evaluated by means of the classical performance metrics applied to regression problems: mean absolute error, mean absolute percentage error, and root-mean-square error. In addition, the imbalance of LOS samples compels us to find metrics that consider this fact or that we can particularize for each class. Consequently we propose the use of recall values obtained from confusion matrices, and discuss other metrics that we can extract from them.

The initial resulting performance was then improved by adding data from all the stretches in the network, as in [6] and [7], and segmenting the training and validation sets considering the specific type of day (Monday, Tuesday-Wednesday-Thursday, Friday, Saturday-Sunday-holiday). For the validation and construction of this predictor we used 12 months of empirical data captured by Bluetooth identifiers deployed forming 6 stretches on an urban highway in Seville (Spain).

II. RELATED WORK

Any traffic prediction model must face its underlying stochastic character [8]. This issue can be faced by the use of either macroscopic/microscopic traffic modeling [9] or empirical approaches based on artificial intelligence. Macro/micro models rely on the careful inclusion of all relevant data, specifically the flows at all ingress and egress points, as well as some details about the road structure and capacity. This often complicates the problem, via the sheer magnitude of the structural representation. For its part, black boxes inadvertently encode these structures into their internal representation, without the need to explicitly represent them. On the negative side, this may hide cause-effect relations. On the positive side, they provide high predictive capabilities even when only partial data is available. Our work focuses on the exploitation of Bluetooth Monitoring Systems (BTMSs) data, which cannot provide the accurate vehicle counting required for the former.

Works based on artificial intelligence adopt two fundamental approaches: parametric and non-parametric. The former relies on the analysis of time series like the auto-regressive integrated moving average [10], and the use of Kalman filters to adapt predicted states to actual observations [11]. On the other hand, non-parametric models seek for patterns among historical data that mimic current traffic situations. The fundamental methods are the k nearest neighbors as in [12] and [6], and neural networks like [13]–[15], and [16].

These methods must be fed with TT data, with origin in different sources, mainly inductive loops, FCD, and AVI systems. Speed values collected by inductive loops can be used to estimate and predict TT [17]. In order to do so, point-like speed values must be extrapolated to the whole stretch under study, which reduces the accuracy of the input data and the subsequent predictor. This effect is even more pronounced if the speed values themselves come from yet a previous estimation that considers the measured intensity, occupancy, and some average vehicle length, as in [18]. On top of these

issues, estimated TT from inductive loops must face the fact that these detectors show poor performances in congested scenarios [19].

Consequently, most TT data come from a statistical calculation made upon the individual time each vehicle invest on traversing the stretch. The direct approach is gathering FCD from vehicles, using their GPS positioning system. In order to obtain accurate TT estimations, we need a significant amount of journeys that share the same origin and destination, and which started on the same time interval. In addition to this, it is fundamental that data reflect the actual behavior of traffic, i.e., they are not biased by any specific feature of the selected group of vehicles. Accordingly, data coming from segments of drivers like taxis [20] may not be representative of the overall traffic. Finally, we should not either rely on data provided by third-parties like Google [13] given that they do not guarantee the accuracy of the TT information they supply.

During the past decade, Bluetooth has emerged as a cost-effective technology to generate TT information [21]. Bluetooth is capable of providing highly accurate TT data in freeways [22]. In any case, every TT estimation method needs to carry out a rigorous processing of the raw data collected by the Bluetooth identifiers in order to eliminate outliers and invalid data [12].

Furthermore short-term predictors require that TT data are collected with high temporal resolution given that this directly affects their prediction horizon. Thus, TT data aggregated in 15-minute intervals cannot provide forecasts beyond 15 minutes ([20] and [13]).

TT predictors are evaluated by means of classical metrics. The objective is to reflect their performance mainly in congested scenarios, given that predicting TT in free-flow is straightforward. Thus, an optimal performance evaluation analyzes each traffic situation independently. Otherwise, metrics will reflect diffused values, heavily loaded with free-flow results as this is the most frequent situation. This issue may appear merging separable traffic conditions in time (peak, valley, and night hours [6]) and space (long stretches as in [17] and [23]).

Due to the majority presence of free-flow data, we require a large and varied sampling to guarantee the representativeness of non-parametric models and avoid overfitting [24]. In order to increase this representativeness and the accuracy of the predictors, some works like [25] introduce seasonal parameters [26]. Despite its intrinsic benefits, this leads to higher computational requirements and a costly initial configuration of the model for every use case and subsequent updates.

In addition, TT as a predicted variable cannot be directly used to infer congestion given that it depends on the length of the stretch and the maximum speed allowed. This is the reason why some authors proposed methods to normalize its values as in [12] and [6].

Most of the presented works approach the TT forecast based on traffic data originated on the stretch under analysis. Nevertheless, road networks are connected and so there exists an evident influence of the traffic in one stretch on the others [7] that is exploited in studies like [6].

From the basis of this previous research, our approach includes the following premises:

- We chose LOS as the prediction objective. This way we provide a normalized variable, widely accepted by traffic engineers and managers.
- We selected a RUSBoost decision tree as our predictor given its suitability to deal with unbalanced data.
- The inputs to the RUSBoost predictor are TT values calculated by our own traffic information system, based on detections provided by Bluetooth vehicle identifiers. This system has been validated in a living lab deployed on SE-30 and A-49 highways (Seville, Spain), in real operation since 2015.
- We used a large dataset, which comprises 12 months of data, thus avoiding any seasonal effects.
- We used recall values calculated from confusion matrices for the performance evaluation of the obtained results, which provide specific metrics for each LOS category.

III. TT AND LOS CALCULATION BASED ON EMPIRICAL BLUETOOTH VEHICLE IDENTIFICATION DATA

A. TT Calculation

TT is a statistical variable that indicates the time required to traverse a specific stretch. It represents the overall behavior of vehicles that started (departure TT, DTT) or finished (arrival TT, ATT) the trip on the same time interval, moving at the speed allowed by traffic conditions.

These two variables are inherently different as they reflect two separate features of the same process [27]. Under free-flow conditions, ATT and DTT show just slight differences. However, these differences drastically escalate at the onset and end of congestions. This fact can be observed in Fig. 1, which depicts ATT and DTT on a link with a length of 2.3 kilometers. In addition to the evident difference between these variables at the onset of congestion, data show that ATT usually lags behind DTT during a transition of traffic state. This means that the intrinsic nature of ATT reduces its capability of promptly reflecting changes in traffic. Consequently, the objective of TT prediction must aim at DTT as the variable to forecast. This way DTT will report to drivers and road managers the time they expect to invest to cover the stretch before starting it, thus allowing them to take informed decisions.

Nonetheless, assigning a DTT value to a specific time interval implies waiting for every vehicle that started the trip on it, to complete it. Thus DTT is inherently a *delayed* variable. On the contrary, ATT does not show this issue as its calculation only needs to look *back* in time and find the starting instants of every vehicle that reached the destination on the time interval under study. This means that ATT can be obtained in real time. However, ATT is not an informative variable for drivers and managers as it does not reflect the *forthcoming* but the *past* traffic conditions.

Consequently, a real-time TT predictor will generate DTT forecasts based on ATT values. The goal is to reproduce the real DTT that we will be able to measure once the individual TTs registered by every vehicle are available. This resulting

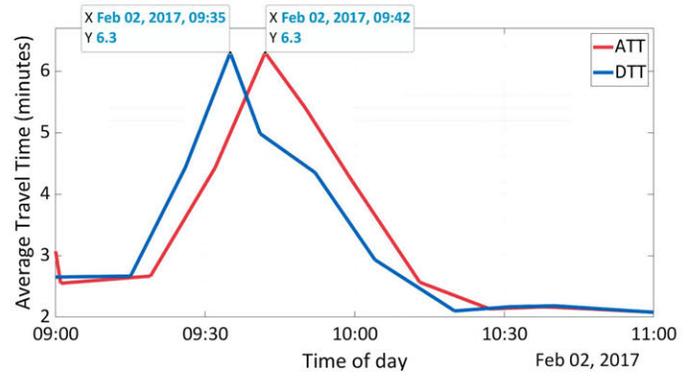


Fig. 1. Comparison between ATT and DTT values.

DTT will act as the gold standard for the evaluation of the predicted results.

B. LOS Calculation

LOS is a discrete variable with 6 different categories ranging from A (free-flow) to F (heavy congestion) as defined in the Highway Capacity Manual (HCM) [28]. LOS is calculated imposing a set of thresholds to the relation between current and optimal traffic conditions. According to the HCM there are different ways to obtain LOS depending on the input variables (flow, speed, TT, etc.) and the road type. In this work we apply the thresholds specified for the *through-vehicle travel speed* in the HCM. This is an average speed that reflects the set of traffic conditions that influence the TT experienced along the stretch. The variable taken as the reference is the *Base Free-flow Speed*, which expresses the maximum speed during “low-volume conditions”. The ratio between these two variables provides a measurement of the performance of the stretch. The actual values are then discretized into a set of 6 ranges corresponding to the different LOS categories. In our case, we take the inverse of the thresholds defined in the HCM in order to convert them into TT-based percentages. This results on the values and descriptions indicated in Table I.

LOS normalizes the absolute values of TT, allowing a direct comparison of traffic conditions between different stretches, independent of their length and configuration.

C. TT and LOS Data From a Bluetooth AVI System in Real Operation

In order to construct and validate the proposed short-term LOS predictor, we used traffic data provided by an AVI system composed by Bluetooth vehicle identifiers. This system is deployed on SE-30 and A-49 highways in Seville (Spain). It consists of 4 unequally spaced Bluetooth detectors, installed on information panels (one per node). These nodes define 6 links; 3 in each direction. Fig. 2 shows the locations of each Bluetooth node and the links they form. In addition, Table II provides the details about the length of every link (in meters) and its corresponding Base Free-flow TT (in seconds).

For the current study we used data from the 12 months in year 2017. The dataset included more than 100 million Bluetooth detections. These detections were processed following

TABLE I
LOS CATEGORIES

LOS	A	B	C	D	E	F
Percentage over Base Free-flow TT	$\leq 118\%$	(118%–149%)	(149%–200%)	(200%–250%)	(250%–333%)	$> 333\%$
Description	Free-flow operation	Reasonably unimpeded operation	Stable operation	Approaching unstable operation	Unstable operation and significant delay	Congestion



Fig. 2. Nodes and links in the AVI system deployed in Seville (Spain).

TABLE II
LINKS IN THE AVI SYSTEM DEPLOYED IN SEVILLE

Link	Origin	Destination	Length (m)	Base Free-flow
				TT (s)
Link 1	Node 1	Node 2	5 854	322
Link 2	Node 2	Node 1	5 887	325
Link 3	Node 2	Node 3	2 332	122
Link 4	Node 3	Node 2	2 369	116
Link 5	Node 3	Node 4	5 606	254
Link 6	Node 4	Node 3	5 229	237

the algorithm exposed in [3]. This algorithm first filters out every detection that does not correspond to a vehicle, using the dedicated inquiry access code on Bluetooth messages. This guarantees that all the data we use was generated by cars thus corresponding to the traffic conditions we want to predict. Next, the methodology integrates multiple detections in origin and destination. Finally, it removes outliers, which plays a pivotal role in TT estimation accuracy.

This process results in DTT and ATT estimations for every link in a 1-minute basis. Thus, we generated $1440 \times 365 = 525\,600$ ATT and DTT values per link, summing up to 3 153 600 records in total for each variable. As said before, ATT acted as the input data of the LOS predictor; as the ground truth we calculated the actual LOS categories from the corresponding DTT values. Finally, let us remark that the 1-minute basis allowed us to perform short-term predictions with the appropriate horizons.

IV. SHORT-TERM LOS PREDICTOR

A. Problem Statement

The objective of this work is to develop a short-term predictor of LOS, fed with ATT data. As presented in Section III,

these data include the ATT values calculated for each minute and link. Thus, at a given moment in time t , current or past, we aim at predicting the future LOS that vehicles will experiment departing from the origin of the stretch in $t + t'$, being t' the selected prediction horizon. The only inputs to the classifier will be ATT values corresponding to time t , generated in one or more links in the network. As an illustrative example, let us assume that the current time is 09:00 and we have fixed a prediction horizon of $t' = 10$ minutes; then the classifier will take the ATT values calculated from the set of vehicles that reached the destination of each link from 08:59 to 09:00 and forecast the expected LOS of a vehicle that will start the stretch under study at 09:10. We will next proceed to state the mathematical framework for the addressed problem.

Let L be the total number of links in the network. Given a current or past time t , a prediction horizon t' , and a link k in the network ($1 \leq k \leq L$), the objective is to predict the LOS expected for vehicles starting link k at time $t + t'$. In order to perform this LOS forecast, we can use ATTs corresponding to: (i) the link under study (k); (ii) all the L links; or (iii) a subset of l links. Thus the feature space of the problem is defined as the set $X_l = \{ATT^j\}$, $j = 1, \dots, l$, $1 \leq l \leq L$. Accordingly, a point in this feature space is represented by a vector $\vec{x}_i = (ATT_i^1, ATT_i^2, \dots, ATT_i^l)$ in which each component ATT_i^j takes the value of the ATT corresponding to sample i and link j . In our study, each sample corresponds to 1 minute in time during year 2017, thus the total number of samples per link is $s = 525\,600$ and $i = 1, \dots, 525\,600$.

On the other hand, each sample (\vec{x}_i, y_i) includes the class y_i in the set $Y = \{A, B, C, D, E, F\}$ associated with the specific LOS calculated for instant i on the link k under analysis. A number of m samples form the dataset S , with $m = 3\,153\,600$ in our case. Let us remark that there is no direct relation between \vec{x}_i and y_i in sample (\vec{x}_i, y_i) given that ATT and DTT (where we derive LOS from) are fundamentally different.

Using a portion of this dataset S , we must train a predictor that generates a model $H_k^{t'}(\vec{x})$ for each link $k = 1, \dots, L$ and prediction horizon t' . This model takes any new sample \vec{x}_t with ATT data calculated in time t (current or past), and estimate the class $y \in Y$ corresponding to link k and time $t + t'$.

B. Random Undersampling Boost Classifier

Our approach to solve the described problem is based on a RUSBoost classifier [29]. RUSBoost is a hybrid sampling/boosting classification algorithm, especially appropriate for learning from unbalanced training data. Our training data is clearly dominated by free-flow situations (LOS A), with

fewer heavy congestion examples (LOS F). Machine learning techniques fail to efficiently classify this kind of skewed data, but RUSBoost solves the problem by combining sampling and boosting.

The algorithm takes the number of members in the class with the fewest in the training data, N , as the basic unit for sampling. Classes with more members are under-sampled by using only N observations of every class. In other words, if there are P classes, RUSBoost takes a subset of the data with N observations from each of the P classes. This process unavoidably loses information, thus reducing the performance of the subsequent classifier. However, if this process is iterated R times, the training will eventually include any information lost in precedent steps. This is the approach taken by RUSBoost.

RUSBoost runs an iterative process. It is initialized by calculating the weight $D_1(i) = 1/m$ of each sample in the dataset S . In each iteration r , with $r = 1, \dots, R$, RUSBoost performs an undersampling of the overall set S , creating subset S'_r with distribution D'_r . The subset S'_r is used to train a *weak learn* classification algorithm. The result of this iteration is a suboptimal model h_r . This model is applied to the complete sample set S generating outputs $h_r(x_i)$ (for example, a numeric confidence rating). Considering the output for each sample and its weight $D_r(i)$, RUSBoost calculates the pseudo-loss ϵ_r in iteration r . Finally, ϵ_r is used to obtain the weights of each sample in the next iteration $D_{r+1}(i)$.

The final output of the RUSBoost classifier is a model $H'_k(\vec{x})$, specified for every link k and prediction horizon t' .

C. Parametrization and Training

We created three different short-term LOS predictors based on the RUSBoost algorithm. **Predictor₁** took as inputs only ATT values corresponding to the link under study. Thus, $l = 1$ and samples were given by (\vec{x}_i, y_i) , where \vec{x}_i had just one component ATT_i^k . On the other hand, **Predictor₂** included as inputs the ATT values from every link in the network, i.e., $\vec{x}_i = (ATT_i^1, ATT_i^2, ATT_i^3, ATT_i^4, ATT_i^5, ATT_i^6)$. Finally, **Predictor₃** segmented the input data depending on the type of day: Monday (Mon), Tuesday-Wednesday-Thursday (Tu/We/Th), Friday (Fri), and Saturday-Sunday-holiday (Sa/Su/ho) taking ATT values for all the links as input data. Each classifier was trained to generate LOS forecasts with 3 prediction horizons: $t' = 5, 10, 15$ minutes.

These predictors used a standard decision tree learner [30] as the *weak learn* classifier of the RUSBoost algorithm. This classifier was iterated $R = 30$ times in the RUSBoost algorithm. We fixed its learning rate to 0.3. In order reach a good performance whilst avoiding overfitting, we particularized the maximum number of splits of each classifier considering the number of feature variables and samples in the dataset. Thus, we selected 32 splits for **Predictor₁**, 256 for **Predictor₂**, 128 for segments Mon, Fri, and Sa/Su/ho in **Predictor₃**, and 196 for segment Tu/We/Th in **Predictor₃**.

We performed all the experiments using 5-fold cross validation. That is, the datasets were split into 5 partitions, 4 of which were used to train the models, while the remaining

A	491918	6073	861	261	102	12
B	1335	7578	1189	409	181	15
C	235	947	4580	794	399	81
D	42	349	540	2802	554	132
E	1	60	130	202	1905	121
F		7	51	75	172	1482
	A	B	C	D	E	F

Fig. 3. Confusion matrix corresponding to the LOS prediction in Link 4, with a prediction horizon $t' = 5$ minutes.

TABLE III
PRIMARY PERFORMANCE METRICS IN A CONFUSION MATRIX

Recall	Precision	Specificity
$R_c = \frac{m_{cc}}{\sum_{j=1}^N m_{cj}}$	$P_c = \frac{m_{cc}}{\sum_{i=1}^N m_{ic}}$	$Sp_c = \frac{\sum_{i=1}^N \left[\sum_{k=1}^N m_{ik} \right] - m_{ic}}{s - \sum_{h=1}^N m_{ch}}$

partition was used as test data. We carried out 5 independent runs of 5-fold cross validation to eliminate any biasing and overfitting that could occur as a result of the random partitioning process.

D. Performance Metrics

The evolution of traffic prediction from TT to LOS turned the problem into classification. The analysis of the performance of a classifier is usually based on a *confusion matrix* and the set of metrics we can derive from it. A confusion matrix M is an $N \times N$ square matrix, $N = |Y|$ ($N = 6$ in the case of LOS) with *true* and *predicted* classes heading each row and column respectively. Each element $m_{ij} \in M$ with $i, j = 1, 2, \dots, N$, represents the number of times the classifier predicted class j in a situation with true class i . Consequently, the elements on its diagonal (m_{ii}) show correct predictions, while the off diagonal cells represent the misclassified predictions. Confusion matrices are often displayed as a table like the one in Fig. 3, resulting from this study.

From this complete view of the actual results obtained from the classifier we can extract a set of performance metrics. *Accuracy* is frequently used to characterize the overall performance of a classifier. It represents the general success rate, given by the ratio of correct predictions over the number of samples. Accuracy only makes sense as an evaluation metric if the class labels are uniformly distributed. In the presence of unbalanced classes, accuracy is heavily biased by the results obtained on the majority class, thus concealing potential low performances on the rest.

In order to avoid this issue, we must use metrics particularized for each LOS [31]. We can extract three primary measures from a confusion matrix: recall, precision, and specificity. Table III shows the mathematical definition of these metrics. *Recall* and *precision* measure the success in

TABLE IV
AVERAGE RECALL FOR LINK, PREDICTOR, AND LOS

Link	Predictor	A	B	C	D	E	F	Total
Link 1	Predictor ₁	91.6 %	55.1 %	33.8 %	26.1 %	47.3 %	70.4 %	54.0 %
	Predictor ₂	92.3 %	60.6 %	42.8 %	47.7 %	75.2 %	85.7 %	67.4 %
	Predictor ₃	92.4 %	66.9 %	61.4 %	69.9 %	85.3 %	90.4 %	77.7 %
Link 2	Predictor ₁	92.4 %	56.6 %	24.4 %	31.9 %	46.7 %	76.9 %	54.8 %
	Predictor ₂	92.6 %	61.5 %	36.6 %	66.9 %	84.6 %	92.8 %	72.5 %
	Predictor ₃	93.3 %	73.4 %	63.0 %	84.4 %	93.3 %	96.0 %	83.9 %
Link 3	Predictor ₁	94.0 %	43.1 %	49.2 %	25.1 %	27.0 %	72.8 %	51.9 %
	Predictor ₂	93.9 %	43.6 %	44.9 %	57.7 %	80.8 %	90.1 %	68.5 %
	Predictor ₃	94.1 %	51.6 %	56.7 %	76.5 %	90.3 %	92.9 %	77.0 %
Link 4	Predictor ₁	97.9 %	47.8 %	41.9 %	34.9 %	31.3 %	72.0 %	54.3 %
	Predictor ₂	98.1 %	62.3 %	60.9 %	67.8 %	77.2 %	81.2 %	74.6 %
	Predictor ₃	98.2 %	79.5 %	81.4 %	86.0 %	84.4 %	88.8 %	86.4 %
Link 5	Predictor ₁	94.9 %	60.1 %	51.7 %	45.1 %	79.9 %	84.6 %	69.4 %
	Predictor ₂	94.0 %	59.4 %	68.1 %	94.3 %	98.2 %	97.5 %	85.2 %
	Predictor ₃	95.2 %	64.8 %	80.5 %	95.5 %	97.4 %	96.7 %	88.3 %
Link 6	Predictor ₁	95.5 %	55.5 %	39.0 %	43.8 %	31.0 %	74.6 %	56.6 %
	Predictor ₂	96.0 %	62.3 %	53.1 %	57.4 %	63.8 %	85.3 %	69.7 %
	Predictor ₃	95.4 %	75.9 %	76.4 %	81.0 %	83.5 %	90.1 %	83.7 %
Total	Predictor ₁	94.4 %	53.0 %	40.0 %	34.5 %	43.9 %	75.2 %	56.8 %
	Predictor ₂	94.5 %	58.3 %	51.1 %	65.3 %	80.0 %	88.8 %	73.0 %
	Predictor ₃	94.8 %	68.7 %	69.9 %	82.2 %	89.0 %	92.5 %	82.8 %

correctly *predicting* class c , thus evaluating m_{cc} against the number of true c samples (recall) and samples predicted as c (precision). In other words, recall focuses on rows whilst precision focuses on columns in the confusion matrix. On the other hand, the *specificity* of class c reflects the capability of the classifier to correctly *discard* samples from other classes.

However, the mere selection of a metric specific to each individual class does not avoid the issue. In the presence of unbalanced classes, column-based performance metrics like precision fail to represent the actual quality of the classifier [32], which was empirically observed in [33]. Observe the column corresponding to the predicted LOS B in Fig. 3. Precision compares the number of correctly predicted LOS B samples (7578) against the total of samples predicted as LOS B (15014; the total sum of that column), resulting in $P_B = 50.5\%$. The summation in the denominator includes the number of true LOS A samples that were classified as LOS B (6073). This number is almost negligible in its row relative to the number of true LOS A samples, but is quite significant in its column. Thus precision is clearly biased by the majority class A, artificially decreasing the resulting performance.

Specificity presents the same issue. However, it creates an opposite effect on performance. In this case, the evident imbalance of LOS A forces the ratio in the calculation of this metric to be very close to 1 despite the class we are analyzing. For example, the specificity of LOS B is $Sp_B = \frac{507452}{514888} = 98.6\%$. Thus specificity arbitrarily increases the resulting performance of the LOS predictor.

On the contrary, recall succeeds in avoiding the issue of the LOS imbalance. It computes the number of successfully predicted samples of a class over the total number of true samples of that same class. This way, the resulting

performance is not sensitive to the uneven distribution of events and reflects the actual quality of the classifier. Consequently, in this work we use recall as the performance metric of the proposed LOS short-term predictor. In addition, we will also provide the average recall (\bar{R}) as a means of directly comparing the overall performance of each predictor.

V. EMPIRICAL RESULTS

We used the dataset of real values of ATT and LOS, captured by the Bluetooth vehicle identifiers deployed on the highway SE-30 in Seville, as inputs for the three LOS classifiers (**Predictor₁**, **Predictor₂**, and **Predictor₃**) described in Section IV. These predictors are presented in a sequence of increasing performances in order to show the impact of the volume and quality of the input data on the overall accuracy of the classifier. These performances are represented by the corresponding recall values specified for each predictor, link, and prediction horizon. As an introductory summary of the results, we present Table IV, which provides the average recalls we obtained for each link, LOS, and predictor, including the three prediction horizons ($t' = 5, 10, 15$ minutes). We will refer to these values during the comparative analysis we perform in the discussions of each particular predictor.

A. LOS Predictor With Input Data From the Same Link

We first present a LOS predictor based on ATT data generated only in the link under study, **Predictor₁**. Table V includes a summary of the obtained results.

Predictor₁ shows an average recall $\bar{R}_1 = 56.8\%$. This low quality is due to the fact that the classifier only manages ATT information from the same link, which is visibly insufficient

TABLE V
RESULTS OF **PREDICTOR**₁: INPUT DATA FROM THE SAME LINK

Delay	Link 1						Link 2					
	A	B	C	D	E	F	A	B	C	D	E	F
5 min.	93.0 %	63.2 %	33.1 %	36.3 %	48.6 %	74.5 %	93.7 %	61.1 %	30.6 %	33.9 %	52.4 %	79.4 %
10 min.	91.3 %	55.8 %	34.9 %	22.7 %	49.6 %	70.6 %	92.7 %	53.1 %	23.5 %	32.6 %	47.9 %	76.7 %
15 min.	90.4 %	46.2 %	33.5 %	19.2 %	43.8 %	66.1 %	90.8 %	55.6 %	19.1 %	29.3 %	39.7 %	74.7 %
Delay	Link 3						Link 4					
	A	B	C	D	E	F	A	B	C	D	E	F
5 min.	95.9 %	50.8 %	54.0 %	30.0 %	32.5 %	74.5 %	98.2 %	57.2 %	51.1 %	43.0 %	37.5 %	74.1 %
10 min.	93.4 %	42.4 %	50.5 %	25.3 %	28.7 %	74.6 %	97.9 %	48.0 %	43.3 %	35.3 %	29.5 %	71.9 %
15 min.	92.7 %	36.0 %	43.2 %	20.1 %	19.9 %	69.2 %	97.5 %	38.3 %	31.3 %	26.5 %	26.9 %	69.9 %
Delay	Link 5						Link 6					
	A	B	C	D	E	F	A	B	C	D	E	F
5 min.	96.2 %	68.0 %	60.7 %	55.6 %	85.3 %	84.3 %	96.4 %	61.7 %	49.1 %	51.0 %	39.5 %	75.1 %
10 min.	94.9 %	60.1 %	51.4 %	45.7 %	79.5 %	83.8 %	95.2 %	56.1 %	39.6 %	41.6 %	31.8 %	74.6 %
15 min.	93.7 %	52.2 %	42.9 %	34.0 %	75.0 %	85.7 %	94.8 %	48.8 %	28.2 %	38.9 %	21.7 %	74.2 %

TABLE VI
RESULTS OF **PREDICTOR**₂: INPUT DATA FROM EVERY LINK

Delay	Link 1						Link 2					
	A	B	C	D	E	F	A	B	C	D	E	F
5 min.	93.4 %	65.3 %	43.6 %	51.3 %	78.6 %	85.9 %	94.0 %	65.5 %	42.1 %	66.1 %	88.2 %	94.5 %
10 min.	92.2 %	59.5 %	43.5 %	47.3 %	76.0 %	85.7 %	92.6 %	60.7 %	35.6 %	68.0 %	83.8 %	92.0 %
15 min.	91.4 %	57.1 %	41.4 %	44.5 %	71.1 %	85.4 %	91.3 %	58.3 %	32.1 %	66.5 %	81.8 %	91.9 %
Delay	Link 3						Link 4					
	A	B	C	D	E	F	A	B	C	D	E	F
5 min.	95.6 %	51.6 %	49.2 %	59.7 %	81.2 %	90.5 %	98.5 %	70.8 %	65.1 %	63.4 %	78.8 %	82.9 %
10 min.	93.6 %	42.1 %	44.1 %	57.0 %	80.0 %	91.5 %	98.1 %	62.7 %	59.9 %	66.6 %	77.3 %	81.1 %
15 min.	92.4 %	37.1 %	41.5 %	56.3 %	81.1 %	88.4 %	97.6 %	53.5 %	57.7 %	73.4 %	75.4 %	79.5 %
Delay	Link 5						Link 6					
	A	B	C	D	E	F	A	B	C	D	E	F
5 min.	95.6 %	65.0 %	73.4 %	95.0 %	98.1 %	97.3 %	97.1 %	68.7 %	58.6 %	63.7 %	62.7 %	86.1 %
10 min.	93.9 %	59.5 %	67.2 %	94.5 %	98.8 %	97.8 %	96.1 %	61.9 %	53.0 %	56.6 %	63.2 %	84.3 %
15 min.	92.4 %	53.6 %	63.8 %	93.5 %	97.6 %	97.3 %	94.8 %	56.4 %	47.8 %	52.0 %	65.5 %	85.4 %

to provide accurate LOS forecasts. However, despite its simplicity, **Predictor**₁ shows a good performance in predicting congestion (LOS F), reaching recall values above 66 % in all cases. Regarding free-flow (LOS A), predictions exhibit high recall values (always above 90 %) as we anticipated, which is not representative of the overall quality of the classifier. Apart from free-flow and congestion, the predictor provides poor performance metrics for the intermediate LOS (B to E). This fact is perfectly sensible as these are *transition* categories showing an intrinsic instability that complicates the prediction task.

In addition, as expected, performance degrades with increasing prediction horizons. It is evident that the correlation between current values of ATT and future LOS decays with time, thus complicating the correct classification.

The performance of **Predictor**₁ was improved making use of the information about ATT in all links as we next present.

B. LOS Predictor With Input Data From All the Links

We built **Predictor**₂ introducing ATT values for all the 6 links in the network as its input data. Table VI shows the recall values we obtained from this predictor.

We can observe an evident improvement in these results that increased the average recall to $\bar{R}_2 = 73.0\%$. **Predictor**₂ reaches a minimum recall of 81.1 % in the classification

of congestion (LOS F), even with prediction horizons of 15 minutes. In addition, it significantly improves the prediction of intermediate LOS, achieving average recalls of 80.0 % and 65.3 % for LOS E and LOS D respectively, and making LOS B and LOS C now exceed 50 %.

The reason for this improvement is twofold. First, we have introduced ATT information from links on the same direction as the stretch under study. Traffic on these stretches has a physical incidence on the others: Upstream congestions alleviate the traffic conditions on subsequent stretches as they reduce the incoming traffic flow; downstream congestions extend to previous stretches, increasing their ATT. Consequently, the predictor can extract valuable information from these ATT data. Second, ATT values on links on the opposite direction may not have a direct physical impact on the traffic we are predicting. However, they carry information susceptible to describe the overall traffic conditions in the network. For example, let us assume that both northbound and southbound links are congested during the morning peak hour, whilst only the latter is congested during the afternoon peak hour and this congestion lasts longer. In this case, the predictor can use the ATT information from the northbound to discriminate whether the congestion of the southbound is *morning-like* or *afternoon-like*, thus increasing the performance of the classifier.

TABLE VII
RESULTS OF **PREDICTOR**₃: DISCRIMINATING BY TYPE OF DAY

Delay	Day	Link 1						Link 2					
		A	B	C	D	E	F	A	B	C	D	E	F
5 min.	All	93.4 %	65.3 %	43.6 %	51.3 %	78.6 %	85.9 %	94.0 %	65.5 %	42.1 %	66.1 %	88.2 %	94.5 %
	Mon	93.4 %	57.8 %	57.3 %	55.7 %	89.1 %	91.6 %	93.1 %	71.6 %	58.7 %	82.5 %	96.0 %	98.1 %
	Tu/We/Th	91.6 %	59.7 %	42.5 %	57.6 %	82.7 %	88.9 %	92.4 %	62.3 %	44.5 %	77.5 %	90.2 %	95.0 %
	Fri	91.9 %	70.9 %	47.4 %	69.4 %	74.2 %	88.2 %	93.6 %	70.8 %	59.2 %	77.6 %	93.3 %	94.3 %
	Sa/Su/ho	98.6 %	90.1 %	98.8 %	100 %	100 %	-	99.1 %	98.7 %	97.8 %	100 %	-	-
10 min.	All	92.2 %	59.5 %	43.5 %	47.3 %	76.0 %	85.7 %	92.6 %	60.7 %	35.6 %	68.0 %	83.8 %	92.0 %
	Mon	92.3 %	54.9 %	56.6 %	56.2 %	90.6 %	94.3 %	91.7 %	67.7 %	56.4 %	83.2 %	95.8 %	98.1 %
	Tu/We/Th	89.1 %	52.6 %	43.2 %	55.5 %	82.2 %	89.0 %	91.4 %	57.0 %	40.4 %	79.0 %	91.5 %	92.1 %
	Fri	90.6 %	65.4 %	48.8 %	70.6 %	71.6 %	89.8 %	91.7 %	67.7 %	56.4 %	83.2 %	95.8 %	98.1 %
	Sa/Su/ho	97.2 %	90.8 %	97.5 %	96.7 %	100 %	-	98.6 %	98.6 %	96.6 %	95.5 %	-	-
15 min.	All	90.4 %	46.2 %	33.5 %	19.2 %	43.8 %	66.1 %	90.8 %	55.6 %	19.1 %	29.3 %	39.7 %	74.7 %
	Mon	91.3 %	54.8 %	54.9 %	57.1 %	88.5 %	95.7 %	9.4 %	66.2 %	57.8 %	80.7 %	95.9 %	98.1 %
	Tu/We/Th	87.8 %	51.0 %	42.4 %	53.9 %	78.4 %	88.1 %	89.4 %	55.0 %	35.0 %	79.1 %	88.6 %	95.2 %
	Fri	89.8 %	64.7 %	49.5 %	66.1 %	66.6 %	87.5 %	90.2 %	67.0 %	54.5 %	75.2 %	92.3 %	94.8 %
	Sa/Su/ho	95.6 %	90.5 %	97.8 %	100 %	100 %	-	98.2 %	97.5 %	99.0 %	100 %	-	-
5 min.	Link 3						Link 4						
	All	95.6 %	51.6 %	49.2 %	59.7 %	81.2 %	90.5 %	98.5 %	70.8 %	65.1 %	63.4 %	78.8 %	82.9 %
	Mon	94.6 %	58.6 %	56.5 %	82.6 %	91.1 %	92.5 %	98.7 %	90.4 %	86.7 %	80.5 %	88.9 %	88.6 %
	Tu/We/Th	93.4 %	47.5 %	55.2 %	73.1 %	93.8 %	93.8 %	98.3 %	71.3 %	70.2 %	61.5 %	80.3 %	87.0 %
	Fri	96.0 %	54.4 %	49.9 %	63.8 %	83.7 %	95.6 %	97.8 %	80.2 %	75.4 %	92.0 %	89.7 %	94.7 %
Sa/Su/ho	98.6 %	66.2 %	72.9 %	89.8 %	94.3 %	83.3 %	99.8 %	92.6 %	100 %	98.9 %	-	-	
10 min.	All	93.6 %	42.1 %	44.1 %	57.0 %	80.0 %	91.5 %	98.1 %	62.7 %	59.9 %	66.6 %	77.3 %	81.1 %
	Mon	92.2 %	51.5 %	54.8 %	79.8 %	87.9 %	94.0 %	98.5 %	87.7 %	87.0 %	85.4 %	86.5 %	89.5 %
	Tu/We/Th	91.9 %	35.4 %	52.9 %	73.6 %	92.8 %	90.4 %	97.8 %	62.8 %	59.2 %	73.5 %	76.4 %	82.5 %
	Fri	94.4 %	47.3 %	48.1 %	63.3 %	83.2 %	94.0 %	96.8 %	75.6 %	78.4 %	85.5 %	90.0 %	93.8 %
	Sa/Su/ho	97.8 %	65.5 %	66.5 %	85.6 %	96.6 %	100 %	99.7 %	91.8 %	100 %	100 %	-	-
15 min.	All	92.4 %	37.1 %	41.5 %	56.3 %	81.1 %	88.4 %	97.6 %	53.5 %	57.7 %	73.4 %	75.4 %	79.5 %
	Mon	90.7 %	49.3 %	55.0 %	79.8 %	91.5 %	91.0 %	98.2 %	85.7 %	86.7 %	86.7 %	87.2 %	88.9 %
	Tu/We/Th	89.2 %	33.2 %	52.2 %	72.7 %	93.5 %	93.2 %	97.4 %	51.5 %	54.6 %	78.6 %	75.7 %	78.9 %
	Fri	93.3 %	45.3 %	47.2 %	61.9 %	85.8 %	92.3 %	95.3 %	73.4 %	83.0 %	89.0 %	85.3 %	95.5 %
	Sa/Su/ho	97.2 %	65.1 %	68.7 %	91.5 %	88.6 %	95.2 %	99.7 %	91.3 %	95.0 %	100 %	-	-
5 min.	Link 5						Link 6						
	All	95.6 %	65.0 %	73.4 %	95.0 %	98.1 %	97.3 %	97.1 %	68.7 %	58.6 %	63.7 %	62.7 %	86.1 %
	Mon	96.1 %	71.8 %	98.7 %	99.7 %	-	-	95.9 %	79.1 %	74.1 %	77.1 %	84.6 %	87.2 %
	Tu/We/Th	95.0 %	62.6 %	72.3 %	96.2 %	98.6 %	93.5 %	96.7 %	68.9 %	72.3 %	68.6 %	65.7 %	90.7 %
	Fri	96.1 %	83.2 %	70.3 %	95.3 %	97.1 %	96.6 %	95.7 %	76.6 %	71.4 %	86.8 %	91.8 %	93.5 %
Sa/Su/ho	98.7 %	62.7 %	75.1 %	100 %	94.4 %	100 %	98.0 %	92.4 %	96.3 %	99.6 %	95.7 %	94.4 %	
10 min.	All	93.9 %	59.5 %	67.2 %	94.5 %	98.8 %	97.8 %	96.1 %	61.9 %	53.0 %	56.6 %	63.2 %	84.3 %
	Mon	94.8 %	67.7 %	97.5 %	99.4 %	-	-	94.3 %	76.0 %	74.7 %	74.7 %	80.6 %	80.6 %
	Tu/We/Th	94.4 %	55.7 %	66.5 %	94.4 %	97.3 %	96.1 %	95.7 %	66.0 %	59.5 %	64.5 %	67.7 %	89.4 %
	Fri	95.1 %	58.2 %	73.5 %	95.7 %	97.6 %	95.4 %	94.3 %	73.3 %	72.7 %	87.4 %	86.5 %	95.9 %
	Sa/Su/ho	98.2 %	75.4 %	79.5 %	84.4 %	94.4 %	100 %	97.5 %	89.6 %	96.6 %	98.2 %	97.7 %	97.2 %
15 min.	All	92.4 %	53.6 %	63.8 %	93.5 %	97.6 %	97.3 %	94.8 %	56.4 %	47.8 %	52.0 %	65.5 %	85.4 %
	Mon	93.1 %	64.9 %	98.2 %	99.4 %	-	-	93.2 %	73.9 %	74.6 %	70.8 %	81.6 %	73.9 %
	Tu/We/Th	90.2 %	47.9 %	68.6 %	94.6 %	97.3 %	94.8 %	94.4 %	54.1 %	58.2 %	58.4 %	68.4 %	86.0 %
	Fri	93.6 %	53.4 %	73.4 %	95.3 %	99.5 %	98.9 %	92.3 %	72.6 %	70.2 %	86.5 %	85.3 %	95.9 %
	Sa/Su/ho	97.0 %	73.4 %	82.5 %	92.2 %	100 %	95.2 %	96.8 %	88.4 %	95.7 %	99.1 %	96.5 %	97.2 %

C. LOS Predictor Discriminating by Type of Day

Traffic often presents different behaviors depending on the specific day in the week. Practitioners know that drivers do not follow the same patterns on a Monday than on a Sunday. We applied this rule to segment the dataset and improve the performance of the classifier, **Predictor**₃. Accordingly,

data samples were separated in 4 categories: Monday (Mon); Tuesday-Wednesday-Thursday (Tu/We/Th); Friday (Fri); and Saturday-Sunday-holiday (Sa/Su/ho). We trained and validated the predictors within these 4 groups. Table VII shows the obtained results; the information of each Link is preceded by the performance of **Predictor**₂ (noted by "All") in order to

TABLE VIII
 PREDICTOR IMPORTANCE ($\times 10^{-3}$) FOR LINK 4 AND PREDICTION HORIZON $t' = 5$ MINUTES

Predictor	ATT Link 1	ATT Link 2	ATT Link 3	ATT Link 4	ATT Link 5	ATT Link 6
Importance	0.112	0.137	0.071	0.793	0.080	0.159

allow a direct comparison. Let us indicate that some of the cases had no samples within year 2017; for instance there were no congestions (LOS E and LOS F) in Link 4 on a Saturday, Sunday or holiday; we note the corresponding cells as “-”.

Predictor₃ improves the previous results, achieving an average recall $\bar{R}_3 = 82.8\%$. The average recall in the prediction of congestion (LOS F) reaches 92.5% and those corresponding to both LOS D and LOS E exceed 80% , whilst the performance in the prediction of LOS B and LOS rises above 68% .

VI. DISCUSSION AND CONCLUSIONS

The work we have presented aims at opening new research lines in the field of short-term traffic prediction. The fundamental change is to evolve from TT to LOS in order to provide practitioners with a meaningful and standard metric to represent traffic conditions despite the length and configuration of stretches. Focusing on LOS moves the problem statement from *regression* to *classification*, thus spreading the range of solutions and tools we have in hand to solve it.

Our analysis has generated a set of observations and questions to be addressed in future works. First, results for the intermediate LOS (B to E) reveal their intrinsic instability, which we could further exploit to detect transitions between traffic states and eventually find the onset and end of congestions. We are currently working on this approach with some initial but promising results that analyze the uncertainty of our predictor and use it as a metric to identify instability points on traffic behavior. In addition, this uncertainty provides a means to improve the weighing algorithms we run during the boosting process and generate detailed metrics about the overall performance.

On the other hand, we have quantitatively demonstrated that traffic shows distinct behaviors on different days in the week and exploited this information. The increase in the performance of **Predictor₃** also highlights the importance of correctly selecting the datasets to train these classifiers. Doing so, we can alleviate the basic tasks of the predictor and allow it to invest its computing potential in building the correlations that we cannot detect by mere inspection.

This observation also leads to question the input data to employ. We have shown that taking the ATT of each link as an input variable enhances the predictor’s performance. We measured the impact of these input variables analyzing their *predictor importance*. Table VIII shows the results we obtained for **Predictor₂** applied to Link 4 with a prediction horizon $t' = 5$ minutes. We can observe that the main predictor is the ATT recorded on the same link under analysis (Link 4), followed by the values of ATT in the two links on its direction (Link 2 and Link 6), and finally the remaining 3 links on the opposite direction (Link 1, Link 3, and Link 5). This observation suggests extending the input set to additional traffic

variables like flow or occupancy, or even road features like the presence of reversible lanes [34]. Nevertheless, we must first somehow state their potential to increase the resulting performance. Consequently, we have started specific work on the analysis of the predictive capabilities of variables, in order to select those that will contribute to a significant improvement on the classifier.

The presented BTMS-based short-term LOS classifier is capable of predicting congestions with an average recall of 92.5% . This predictor shows an overall performance of 82.8% with prediction horizons of up to 15. The classifier was trained and validated using 12 months of real traffic data provided by a BTMS deployed on the SE-30 highway in Seville (Spain). This validation included 5 independent runs of a 5-fold cross validation. We are currently applying this predictor to an urban network in the city of Madrid (Spain) in order to optimize traffic lights. Moreover, we are integrating this technology in a congestion pricing tool for toll roads.

In view of the foregoing, BTMS-based LOS prediction appears as a promising approach to traffic forecasting, opening interesting opportunities for researchers and practitioners.

ACKNOWLEDGMENT

The authors would like to thank the contributions and support of Sociedad Ibérica de Construcciones Eléctricas (SICE) and, specifically, Beatriz García and Tomás Paadín.

REFERENCES

- [1] I. Lana, J. Del Ser, M. Velez, and E. I. Vlahogianni, “Road traffic forecasting: Recent advances and new challenges,” *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 2, pp. 93–109, 2018.
- [2] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, “Short-term traffic forecasting: Where we are and where we’re going,” *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 3–19, Jun. 2014.
- [3] J. J. Vinagre Diaz, A. B. R. Gonzalez, and M. R. Wilby, “Bluetooth traffic monitoring systems for travel time estimation on freeways,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 123–132, Jan. 2016.
- [4] R. Roess, “Level of service concepts: Development, philosophies, and implications,” *Transp. Res. Rec.*, no. 971, pp. 1–6, Jan. 1984.
- [5] C. Oh, A. Tok, and S. G. Ritchie, “Real-time freeway level of service using inductive-signature-based vehicle reidentification system,” *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 138–146, Jun. 2005.
- [6] P. Cai, Y. Wang, G. Lu, P. Chen, C. Ding, and J. Sun, “A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting,” *Transp. Res. C, Emerg. Technol.*, vol. 62, pp. 21–34, Jan. 2016.
- [7] S. R. Chandra and H. Al-Deek, “Predictions of freeway traffic speeds and volumes using vector autoregressive models,” *J. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 53–72, May 2009.
- [8] B. Van Arem, H. R. Kirby, M. J. M. Van Der Vlist, and J. C. Whittaker, “Recent advances and applications in the field of short-term traffic forecasting,” *Int. J. Forecasting*, vol. 13, no. 1, pp. 1–12, Mar. 1997.
- [9] H. Payne, “Models of freeway traffic and control,” *Math. Models Publ. Sys.*, vol. 28, pp. 51–61, Jan. 1971.
- [10] J. Wang, I. Tsapakis, and C. Zhong, “A space–time delay neural network model for travel time prediction,” *Eng. Appl. Artif. Intell.*, vol. 52, pp. 145–160, Jun. 2016.

- [11] A. Ladino, A. Kibangou, H. Fourati, and C. C. de Wit, "Travel time forecasting from clustered time series via optimal fusion strategy," in *Proc. Eur. Control Conf. (ECC)*, Jun. 2016, pp. 2234–2239.
- [12] H. M. Moonam, X. Qin, and J. Zhang, "Utilizing data mining techniques to predict expected freeway travel time from experienced travel time," *Math. Comput. Simul.*, vol. 155, pp. 154–167, Jan. 2019.
- [13] F. Goudarzi, "Travel time prediction: Comparison of machine learning algorithms in a case study," in *Proc. IEEE 20th Int. Conf. High Perform. Comput. Commun., IEEE 16th Int. Conf. Smart City, IEEE 4th Int. Conf. Data Sci. Syst. (HPCC/SmartCity/DSS)*, Jun. 2018, pp. 1404–1407.
- [14] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transp. Res. C, Emerg. Technol.*, vol. 54, pp. 187–197, May 2015.
- [15] A. Csikos, Z. J. Viharos, K. B. Kis, T. Tettamanti, and I. Varga, "Traffic speed prediction method for urban networks—An ANN approach," in *Proc. Int. Conf. Models Technol. Intell. Transp. Syst. (MT-ITS)*, Jun. 2015, pp. 102–108.
- [16] J. Rupnik, J. Davies, B. Fortuna, A. Duke, and S. S. Clarke, "Travel time prediction on highways," in *Proc. IEEE Int. Conf. Comput. Inf. Technol., Ubiquitous Comput. Commun., Dependable, Autonomic Secure Comput., Pervasive Intell. Comput.*, Oct. 2015, pp. 1435–1442.
- [17] 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.
- [18] J. Rice and E. vanZwet, "A simple and effective method for predicting travel times on freeways," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 3, pp. 200–207, Sep. 2004.
- [19] M. R. Wilby, J. J. V. Díaz, A. B. R. González, and M. Á. Sotelo, "Lightweight occupancy estimation on freeways using extended floating car data," *J. Intell. Transp. Syst.*, vol. 18, no. 2, pp. 149–163, Apr. 2014, doi: [10.1080/15472450.2013.801711](https://doi.org/10.1080/15472450.2013.801711).
- [20] E. Jenelius and H. N. Koutsopoulos, "Urban network travel time prediction based on a probabilistic principal component analysis model of probe data," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 436–445, Feb. 2018.
- [21] J. S. Wasson, J. R. Sturdevant, and D. M. Bullock, "Real-time travel time estimates using media access control address matching," *ITE J. Inst. Transp. Eng.*, vol. 78, no. 6, pp. 20–23, Jun. 2008.
- [22] A. Stevanovic, C. L. Olarte, Á. Galletebeitia, B. Galletebeitia, and E. I. Kaiser, "Testing accuracy and reliability of MAC readers to measure arterial travel times," *Int. J. Intell. Transp. Syst. Res.*, vol. 13, no. 1, pp. 50–62, Jan. 2015, doi: [10.1007/s13177-014-0081-4](https://doi.org/10.1007/s13177-014-0081-4).
- [23] A. Simroth and H. Zable, "Travel time prediction using floating car data applied to logistics planning," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 1, pp. 243–253, Mar. 2011.
- [24] W. Qiao, A. Haghani, and M. Hamedi, "A nonparametric model for short-term travel time prediction using Bluetooth data," *J. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 165–175, Apr. 2013, doi: [10.1080/15472450.2012.748555](https://doi.org/10.1080/15472450.2012.748555).
- [25] M. Yang, Y. Liu, and Z. You, "The reliability of travel time forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 162–171, Mar. 2010.
- [26] Y. Kamarianakis, H. Oliver Gao, and P. Prastacos, "Characterizing regimes in daily cycles of urban traffic using smooth-transition regressions," *Transp. Res. C, Emerg. Technol.*, vol. 18, no. 5, pp. 821–840, Oct. 2010.
- [27] J. Kim, J. Rho, and D. Park, "On-line estimation of departure time-based link travel times from spatial detection system," *Int. J. Urban Sci.*, vol. 13, no. 1, pp. 63–80, Jun. 2009, doi: [10.1080/12265934.2009.9693646](https://doi.org/10.1080/12265934.2009.9693646).
- [28] *Highway Capacity Manual 6th Edition: A Guide for Multimodal Mobility Analysis*, 6th ed, TRB Publications, Washington, DC, USA, 2016.
- [29] C. Seiffert, T. M. Khoshgoftar, J. Van Hulse, and A. Napolitano, "RUSBoost: A hybrid approach to alleviating class imbalance," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 40, no. 1, pp. 185–197, Jan. 2010.
- [30] L. Breiman, J. Friedman, C. J. Stone, and R. Olshen, *Classification Regression Trees*, 1st ed. Boca Raton, FL, USA: CRC, 1984.
- [31] A. Tharwat, "Classification assessment methods," *Appl. Comput. Informat.*, to be published.
- [32] T. Fawcett, "Introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, pp. 861–874, Jun. 2006.
- [33] D. Ballabio, F. Grisoni, and R. Todeschini, "Multivariate comparison of classification performance measures," *Chemometric Intell. Lab. Syst.*, vol. 174, pp. 33–44, Mar. 2018.
- [34] J. R. D. Frejo, I. Papamichail, M. Papageorgiou, and E. F. Camacho, "Macroscopic modeling and control of reversible lanes on freeways," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 948–959, Apr. 2016.



Mark Richard Wilby received the Ph.D. degree in physics from Imperial College London, London, U.K. He was a Lecturer with the Department of Electronic and Electrical Engineering, University College London, U.K., the Universidad Carlos III de Madrid, Madrid, Spain, and Universidad Rey Juan Carlos, Madrid. He was the Chief Technology Officer and the Chief Executive Officer of several companies. He is currently with the Department of Mathematics Applied to Information and Communication Technologies, Universidad Politécnica de Madrid, Madrid. His research interests include processing, understanding, and control of distributed data and sensor systems.



Ana Belén Rodríguez González was born in Plasencia, Cáceres, Spain. She received the B.S. degree in telecommunication engineering from the Universidad de Valladolid, Valladolid, Spain, in 2000, the B.A. degree in economics from the Universidad Nacional de Educación a Distancia, Madrid, Spain, in 2002, and the Ph.D. degree in telecommunication engineering from the Universidad Carlos III de Madrid, Madrid, in 2008. She was a Professor with the Universidad Carlos III de Madrid and Universidad Rey Juan Carlos, Madrid. She is currently with the Department of Mathematics Applied to Information and Communication Technologies, Universidad Politécnica de Madrid. Her research interests include data analytics, wireless sensor networks, energy efficiency, and intelligent transportation systems.



Rubén Fernández Pozo received the B.S. and Ph.D. degrees in telecommunications engineering from the Universidad Politécnica de Madrid (UPM), Madrid, Spain, in 2003 and 2011, respectively. He was an Assistant Professor in Master in Signal Theory and Communications (MSTC), UPM. He is currently working with the Department of Mathematics Applied to Information and Communication Technologies, UPM. His research interests include machine learning, data analysis and signal processing related to spoken language technologies, intelligent transportation systems, and human-machine interaction.



Juan José Vinagre Díaz received the B.S. degree from the Universidad Politécnica de Madrid (UPM), Madrid, Spain, in 1998, and the Ph.D. degree from the Universidad Carlos III de Madrid in 2005, all in telecommunications engineering. He was a Professor with Universidad Rey Juan Carlos and the Universidad Carlos III de Madrid. He is currently with the Department of Mathematics Applied to Information and Communication Technologies, UPM. His research interests include intelligent transportation systems, energy efficiency, and integration platforms.