This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



# How Smartphone Accelerometers Reveal Aggressive Driving Behavior?—The Key Is the Representation

Manuel Ricardo Carlos, Luis C. González, *Member, IEEE*, Johan Wahlström, Graciela Ramírez, Fernando Martínez, and George Runger

***Abstract*— Aggressive driving behavior is one of the leading causes of road accidents worldwide. One way to ameliorate this situation is to collect and analyze driving patterns with the intention of promoting driving awareness, a task that has been called driving analytics (DA). DA employing the driver's smartphone has attracted attention from the community given its good capabilities to capture, via its integrated sensors, data that could be exploited to infer driving style. Most works in the related literature have represented this sensor information either as statistical scores or in raw format that are later fed into threshold-based heuristics or machine learning (ML) approaches. Based on the hypothesis that better data representations do exist, in this paper, we propose a second-order representation, based on the bag-of-words' strategy, to model accelerometer timestamps associated with aggressive driving maneuvers. We evaluate this representation in two scenarios and three data sets against the best reported work in each of them. In the first scenario, we classify accelerometer samples as either belonging to aggressive or safe driving style. In the second scenario, we approach a multi-class problem, where we are now interested in identifying the exact aggressive maneuver that the accelerometer sample represents. The results show that this novel representation outperforms both state-of-the-art works with 6% and 15% in F-measure for each scenario, respectively. To further investigate the strength of our representation, we make a comparison against similar second-order strategies that have also proved to be successful. Overall, this analysis suggests that this representation constitutes an attractive method for driving behavior classification, boosting discriminative performance of ML approaches.**

***Index Terms*— Driving analytics, aggressive driving, driving behavior, bag of words, insurance telematics.**

## I. Introduction

WHO has not witnessed an Aggressive Driving maneuver? This rhetorical question only shows how pervasive this behavior is among drivers. Unfortunately, aggressive driving maneuvers are frequent and, sometimes, systematically performed. In fact, aggressive driving behavior is one of the main causes of car accidents worldwide which, in turn, represent thousands of deaths per year. Consider for example that just speeding, which is one of the most common aggressive driving actions, killed 15,479 drivers in 2014 in the U.S. alone.[1]

The analysis of driving behavior originally started in the social sciences [1]–[3]. Just recently, with the emergence of affordable sensing and computing platforms, new opportunity areas have been identified [4]–[6]; one being the analysis of driving performance through the use of mobile technology, a field also known as Smartphone Driving Analytics [7]. Driving analytics has attracted plenty of attention from the community since it has been shown that driving behavior can be inferred from data gathered by smartphone sensors [8]. Moreover, positive feedback could be given to the driver thus helping improve the planning and execution of safe maneuvers [4].

The rationale for the appearance of Smartphone Driving Analytics is logical. As mobile technology matured, smartphones grew in technological capabilities, getting their functionality enriched with a great variety of sensors. This phenomenon allowed the smartphone to become an ad-hoc instrument to sense contextual variables. One case of particular interest is when the dynamics of the vehicle are measured using inertial sensors [4], [8]–[15]. These studies have to a large extent been based on accelerometer measurements, and it has been established that accelerometers are one of the most important sensors in smartphone-based driving inference. Notwithstanding, works in the literature have done little to find new ways to better exploit what the accelerometers capture, most of the time resorting to only using simple statistical scores, i.e. mean, variance, or even raw data.

The problem of assessing driver behavior is typically solved in three steps. First, the sensor readings are preprocessed to find candidate signals that could describe the driving behavior in an informative manner. Second, the candidate signals are used to classify the extracted driving events. This can either be a binary classification (aggressive or safe), or a multi-class classification where multiple driving events (or maneuvers) such as braking, cornering, and swerving are identified. Third, the distribution of the driving maneuvers is analyzed and used to inform the driver via a driving score or other type of report his performance at the wheel.

Manuscript received August 25, 2018; revised April 17, 2019; accepted June 21, 2019. The work of M. R. Carlos and L. C. González was supported by Google Inc., under a Latin American Research Award. The Associate Editor for this paper was L. Li. *(Corresponding author: Luis C. González.)*

M. R. Carlos, L. C. González, G. Ramírez, and F. Martínez are with the Facultad de Ingeniería, Universidad Autónoma de Chihuahua, Chihuahua 31125, Mexico (e-mail: lcgonzalez@uach.mx).

J. Wahlström is with the Department of Computer Science, University of Oxford, Oxford OX1 2JD, U.K.

G. Runger is with the School of Computing, Informatics and Decision Systems Engineering, Arizona State University, Tempe, AZ 85281 USA.

Digital Object Identifier 10.1109/TITS.2019.2926639

[1] http://www.iihs.org/iihs/topics/t/general-statistics/fatalityfacts/gender

The main contribution of this study is the introduction of a novel representation for accelerometer data that could be exploited in the context of Driving Analytics. This representation is of second-order and it is based on Bag of Words (BoW) model [16] which, to the best of our knowledge has not been used in this context. One of the advantages of the BoW model is that it can, by itself, learn highly discriminative representations of the data, that are not possible to find by a human observer. To evaluate the BoW model, we took as case studies two scenarios of the second task of the problem of assessing driver behavior (described above) namely, the detection and classification of aggressive driving events. In the first case study we took the recent work of Žylius [17], in which up to 78 features obtained from the accelerometer signal in both time and frequency domain are employed to represent aggressive and safe driving style. Afterwards, for the second case study we consider the recent work of Júnior *et al.* [14], where the authors represent different aggressive driving maneuvers by a feature vector composed of statistical scores of the accelerometer signal. For this case, we approach a multi-class classification, where the goal is to unambiguously identify individual examples of aggressive driving maneuvers.

To further investigate the strength of the BoW representation, we add another experiment to this study, where we contrast our proposal against models that are similar *in spirit*: Bag of Patterns [18], and a Bag of Words over a Symbolic Aggregate Approximation (SAX) representation [19]. Given the results that this representation achieves, and although it is out of the scope of this work, this proposal opens up the possibility of new experiments, where an interested reader could attempt to use it to model data obtained by other inertial sensors, e.g. gyroscopes, to boost knowledge extraction in DA contexts.

The rest of this document is structured as follows: Section II presents an overview of the relevant literature, and Section III describes the data sets that represent both case studies. Section IV specifies the methodology and performance metrics. Section V presents results, organized in three different experiments, and finally, Section VI presents the conclusions and future work.

## II. Related Work

Three main approaches to smartphone-based driver behavior classification can be discerned from the literature. These are dynamic time warping (DTW), threshold-based approaches, and machine learning methods [13]. In this section, we give a brief review of previous work within all of these. Interestingly, some works have reached a real application of the technology,[2] however, it is hard to contextualize their limitations and challenges because of their proprietary nature.

Accelerometers stand out as the most commonly employed sensor within smartphone-based driver behavior classification. For example, Johnson and Trivedi [8] performed end-point detection on accelerometer data to find out when an event of interest has occurred. The event is then classified as a normal or aggressive turn, acceleration, braking, or swerving, by means of a DTW comparison with predefined templates for each class. A similar procedure is followed by Eren *et al.* [11], applying the algorithm of end-point detection on an energy metric calculated from accelerometer readings to detect possible events, which are then classified with DTW. The detections are then passed through a Bayesian classifier to classify the driving style as safe or dangerous. Dai *et al.* [10] attempted to detect drunk driving by finding sudden changes in direction and speed. They used a threshold filter applied to lateral and longitudinal acceleration to identify such events. Fazeen *et al.* [9] also explored thresholds on accelerometer readings to detect gear shifts, lane changes, and sudden braking and accelerating. Eboli *et al.* [20], [21] recently proposed a threshold-based methodology to analyze the relationship between acceleration (lateral and longitudinal) and speed. To employ this characterization the authors used a g-g diagram that serves as guide to intuitively show if a vehicle is being driven in a aggressive or safely manner. Threshold strategies, although simple and intuitive, still present several challenges to overcome, for instance, finding one threshold value that yields good results under most conditions is not easy when processing accelerometer data, given the multitude of variable conditions (location of the sensors, individual driving style, state of the road and traffic flow, etc.) [22].

Machine Learning (ML) is another approach that has been attempted to offer a solution for this task. For example, Vlahogianni and Barmpounakis [7] addressed driving maneuver classification by employing the MODLEM algorithm, which finds the minimal set of decision rules to maximize the discriminative power to detect harsh acceleration, braking, and cornering. Predic and Stojanovic [23] worked on the detection of lane changes, obstacle avoidance, and harsh braking by using decision trees on feature vectors that contain statistical and signal processing metrics. Recently, Žylius [17] explored the usage of histogram features, correlation coefficients, data threshold validation, jerk profile, and spectral information to classify safe and aggressive driving using accelerometer data in the time and frequency domain. By selecting the best six features and applying a random forest classifier, an accuracy of up to 95.5% was achieved. Júnior *et al.* [14] evaluated different ML approaches applied to data from several sensors to find the best discriminative model for a multi-class classification problem, concluding that the usage of combinations of acceleration and angular velocity represented by statistical summarization can produce results with over 0.98 in mean area under the receiver operating characteristic (ROC) curve. Since the application of ML approaches has yielded the best reported scores, it was just a matter of time before the first ensemble classifiers appeared. Reference [24] proposed an ensemble of classifiers to detect dangerous and safe maneuvers, reporting an F1 score of about 93%. The ensemble is integrated by a decision tree, a multi-layer perceptron, a support vector machine and a k-nearest neighbor classifier.

As can be seen, Machine Learning approaches have become the *de facto* machinery to tackle this problem, this caused by their competitive results, outperforming classical algorithms such as those based on thresholds or DTW techniques [25]. Notwithstanding, all of these proposals have fed their models

[2] Aviva Drive, Greenroad, Ingenie, Snapshot and SeeingMachines.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



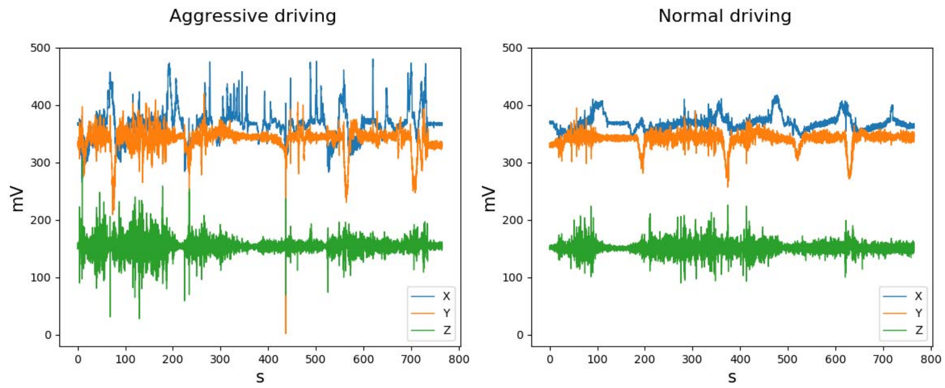



Fig. 1. Raw values from Žylius' data set for aggressive and normal driving, in the original units (millivolts). The values represent lateral (X), longitudinal (Y), and vertical (Z) acceleration.

with accelerometer data represented by simple statistical scores or in some cases, raw format. In this work we propose to advance in the analysis of this problem by proposing a second-order representation for this same data. To evaluate this representation we will revisit the state-of-the-art works where publicly available dataset are introduced, these being the works of Žylius [17] and Júnior *et al.* [14].

## III. Data Sets

To evaluate our proposal, we will use three data sets composed of events of driving behavior collected using accelerometers. This section describes what information each data set has and what its purpose is.

### A. Žylius Data Set

This data set, collected in [17] and kindly provided to us by the author, is made of ten times series (TS) containing about 2.6 hours of triaxial accelerometer readings (in total) at a sampling rate of 17 Hz. Six TS were captured while driving aggressively while the other four when driving in a normal (not explicitly aggressive) manner. Accordingly, each time series was tagged as *normal* or *aggressive*.[3] This data set does not have information about the exact type of aggressive maneuvers that were performed, we only know that each TS represents a driving session where the driver aimed to assume an aggressive or safe driving style. Consider Figure 1, which presents examples of aggressive and normal driving style, to offer a glimpse of this data.

In order to prepare this data to be fed into the classification algorithms, we followed the same procedure reported in the original article. That is, these TS were pre-processed by using unsupervised learning to remove redundant data and extract windows of 1,000 observations, corresponding to one minute of data, with 50% overlap. This process removes segments of the signal in which there is little evidence that the vehicle is moving, by clustering small segments based on metrics of the trends in acceleration, and the energy in low frequencies.[4] After this process was applied, out of the original six "aggressive" time series, 65 windows labeled as aggressive were obtained, whereas for normal driving (which originally consisted in four times series), 15 windows were derived. Note that this procedure yields an unbalanced data set. This data set will be used to evaluate the ability of the methodology to discriminate aggressive from normal driving (Experiment 1).

### B. Ferreira et al.'s Data Set

This data set was made publicly available by Júnior *et al.* [14]. Data was collected with a a Motorola smartphone, using the accelerometer, magnetometer, gyroscope, and a virtual linear accelerometer. The authors report a sampling rate between 50 and 100 Hz, depending on the sensor, and from their data it is possible to determine that they employed a sampling rate close to 50 Hz for the accelerometers. Their experiment was performed in four car trips of approximately 13 minutes each, in which two different drivers completed these trips, ending up with 104 minutes of driving data. The length of collected driving events are accommodated in the range from 2 to 7 seconds depending on how aggressive the event is, according to [14]. The vehicle used to collect the signals was a 2011 Honda Civic, and the smartphone was placed in a fixed position, without being neither moved nor operated while data collection was performed. The types of driving events and the corresponding number of collected events are: aggressive braking (12), aggressive acceleration (12), aggressive left turn (11), aggressive right turn (11), aggressive left lane change (4), aggressive right lane change (5) and non-aggressive event (14). In total, there are 69 driving events categorized into seven different types. Although this data set is small regarding the number of total aggressive driving events, it is one of the most complete with respect to the types of aggressive driving maneuvers that are considered in the literature. This data set will serve to evaluate the methodology in the scenario

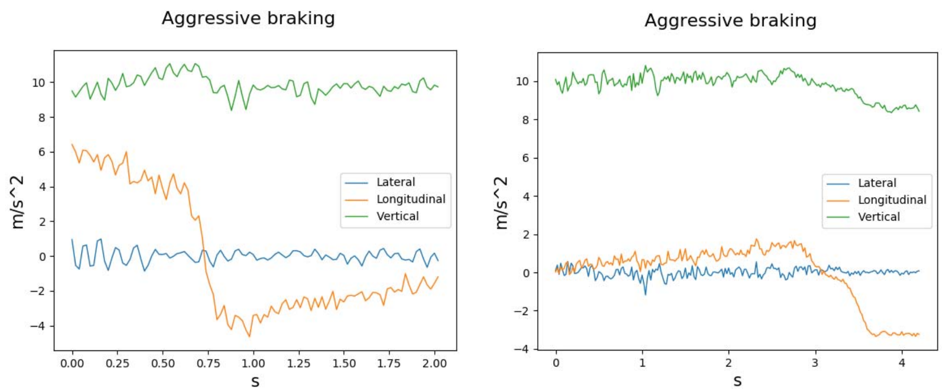
[3]Consider that several aggressive driving maneuvers could have been performed during an aggressive driving session, so that each driving maneuver could have different length.

[4]The Augmented Dickey-Fuller (ADF) statistic, Akaike Information Criterion (AIC), Principal Component Analysis (PCA), Gaussian Mixture Models (GMM), and Median and Mean filters described by Žylius were implemented at this step.

TABLE I

INDIVIDUAL ANOMALIES BY TYPE IN THE AUTHORS' DATA SET. TIME SERIES (TS) LENGTH IS SPECIFIED BY TIMESTAMPS FROM THE ACCELEROMETER SIGNAL

| Driving event | # of driving events | TS length | Avg length | Stdev. of length |
|---|---|---|---|---|
| Swerving left | 75 | 48-200 | 109.74 | 28.23 |
| Swerving right | 75 | 62-200 | 111.18 | 24.21 |
| Sudden braking | 150 | 85-200 | 152.29 | 32.15 |
| Sudden acceleration | 150 | 38-200 | 130.22 | 39.87 |
| Total | 450 | | 125.86 | 31.12 |




(a) (b)

Fig. 2. Triaxial acceleration readings for aggressive braking events from (a) Ferreira *et al.*'s, and (b) the authors' datasets.

of multi-class classification, and also to contrast the BoW against similar approaches (experiments 2 and 3, respectively)

### *C. Authors' Data Set*

Motivated by the work of Júnior *et al.* [14], we introduce a third publicly available data set, which can be found at this repository.[5] This data set consists of examples of single aggressive driving maneuvers, specifically: swerving left, swerving right, sudden braking and sudden acceleration. Note that the categories named as *swerving* could contain examples of aggressive lane change, obstacle avoidance and cornering. Table I presents some numbers of this data set. These maneuvers were performed by subjects with between 10 and 15 years of driving experience. In the data collection session, external observers registered the ground truth labels for each event. Data was captured with 2013 Motorola Moto G smartphones with Android OS version 5.1, containing a ST Micro LIS3DH tri-axial accelerometer. Smartphones were freely placed in the driver's door lower compartment and in the cup holder of the vehicle. Note that given the driving maneuvers that were being executed, the smartphones could potentially change their orientation at any given moment, this is meant to replicate a real-world scenario. Two vehicles were used for this data collection: a Honda Accord and a Nissan Altima. The sampling rate of the accelerometer was set 50 Hz, as in [14]. This data set will also serve for experiments 2 and 3. In order to offer a glimpse of Ferreira *et al.*'s and our data sets, we present Figure 2 where an example of aggressive braking from each data set is shown.

[5]https://www.accelerometer.xyz/datasets/

## IV. METHODOLOGY

In this section we explain the crux of our proposal, that is, how the Bag of Words technique could be applied to the accelerometer signal in order to obtain a feature vector to represent aggressive driving maneuvers. We also give some details of the Machine Learning classifiers and the performance metrics that will be used to test our proposal.

### *A. Data Preparation*

As previously stated, using the accelerometer sensor to identify abrupt movements of a vehicle could become a tough task given the great variety of variables that could add noise to the readings in the different sensor's axes, e.g. sensor orientation with respect to the vehicle can be unknown and be changed without warning. In order to correct some of these artifacts a reorientation procedure needs to be applied. We can easily separate the vertical component from the other two axes by using any of the partial reorientation strategies discussed in [26], but it is nontrivial to extract longitudinal and lateral acceleration from sensors with unknown orientations, especially when GPS and gyroscope data are not available [7]. Raw data provided by Žylius was captured with a known alignment between the sensors' axes and those of the vehicle, so no reorientation was needed. Authors' data was acquired without knowledge of the smartphone orientation, therefore we manually corrected the orientation by visually inspecting plots of the time series after performing vertical reorientation. That is, we performed Principal Component Analysis (PCA) [27] over the X and Y axes, centering (on 9.81 for Z, and 0 for the other two axes) and invert X and Y axes, as necessary, so that the shapes of the events correspond to the tag assigned when the maneuver was performed. A similar method for re-orientation of smartphone data had previously been outlined in [28]. Refer to Section IV-A in [13] for a review on methods for smartphone-to-vehicle alignment. Since not all driving events are of the same length, Gaussian noise padding (with the mean and standard deviation matching that of each sample) were applied to each axis to get a uniform length of 200 sample points, which corresponds to a register of 4 seconds commonly reported in literature. In the case of Ferreira *et al.*'s dataset, data were oriented with respect to the vehicle, so that only centering and the padding procedure were applied to each individual event.

### *B. The Bag of Words Model to Detect Aggressive Driving Behavior*

Hand-crafting feature vectors and choosing preprocessing filters is a nontrivial task. Perhaps for this, all reported works attending this problem have focused on tuning the classification algorithms, instead of looking for better representation of the data; assuming that the classifier will *save the day*, even when feature vectors are simply constructed with statistical scores or raw data. An alternative to this is the usage of non-supervised learning techniques, that is, employing algorithms that can learn highly discriminant features from the data even if those features do not appear to be meaningful to a human observer.

One such form of unsupervised feature learning is the Bag of Words approach (BoW) [16], which will produce a *second-order* representation for the data of interest by treating time series of accelerometer readings as text documents. This is done by considering groups of contiguous readings as words, and counting the number of times each word appears in the signal of interest. This method has previously been applied to raw and filtered accelerometer data in biomedical applications [29], and to some other representations for time series [30]. However, it has not been utilized in the context of smartphone-based driver behavior classification. In our experiments, we use the same BoW model described in [12], which yielded very good results for road anomaly detection when considering only one accelerometer axis, $z$, perpendicular to the road. In this work we generalize this idea to the longitudinal and lateral axes, namely, $x$ and $y$, which had not been attempted prior to this study.

The BoW model consists of two stages, illustrated in Figure 3 and described as pseudocode in Algorithms 1 and 2, respectively. The aim (result) of the Fitting stage is to find $K$ sequences that resemble all the subsequences of acceleration readings of the same group, thus finding a *vocabulary* to uniformly represent the time series in the learning problem. These $K$ segments are called *codewords*. Note that, when looking for these codewords, we employed a sliding window of length $L$ to traverse each training signal, and that there exists an overlap of $O$ timestamps between consecutive windows. For a given accelerometer signal $s$, the $SplitInWindows(s, L, O)$ function extracts all segments of size $L$ that are obtained from $s$ with a sliding window with overlap. At the end, with the

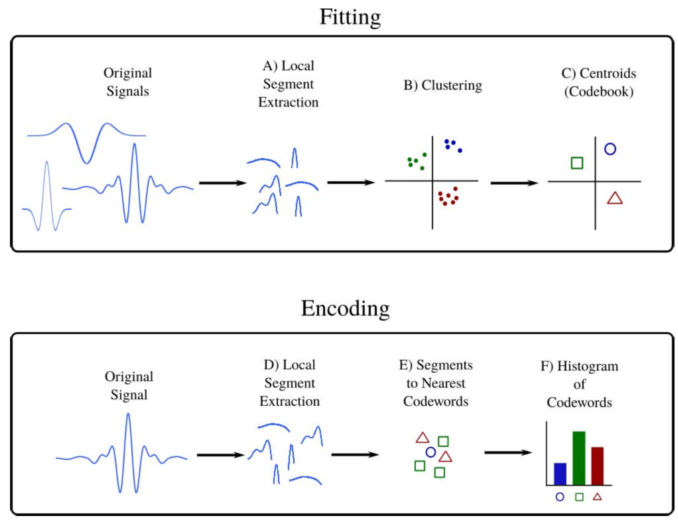



Fig. 3. Application of the two stages of a Bag of Words model. In the first stage (fitting): A) a collection of time series is split in windows of length $L$; B) all segments are considered as points in an $L$-dimensional space, and a clustering algorithm is used to find $K$ clusters; C) The centroids of the clusters are considered as *codewords*, collectively forming a *codebook*. In the second stage (encoding): D) each signal is split in windows of length $L$; E) each window is represented as a codeword by finding the nearest previously determined centroid, converting the time series to a sequence of codewords; F) a histogram is calculated for this sequence of codewords, a vector of $K$ dimensions containing the number of times each codeword is found in the original signal becomes the encoded form of the time series.

list W containing all the signal segments (from all signals in $S$) a clustering algorithm is applied to find the $K$ centroids (codewords). The set of all codewords is called the *codebook*. The codebook can be seen as *building blocks* of the final feature vector.

**Algorithm 1:** Fitting Stage

**Input**: S: list of training signals of the same classes; K: number of codewords to find; L: length of sliding window; O: number of timestamps that overlap between consecutive windows.
**Result**: $K$ codewords to represent the class.
W ← {} // Segments from the original signals.

**foreach** $s$ *in* $S$ **do**
  W ← W + SplitInWindows(s,L,O)
**end**
codewords ← K-Means(W,K)
**return** $K$ codewords

The second stage, Encoding, constructs the feature vector of an accelerometer signal sample $s$. First, it extracts all segments of size $L$ in the same way as in the Fitting stage. For each of these segments it finds the closest codeword within the codebook using an Euclidian distance in an $L$-dimensional space. Once the closest codeword is found, its count increases by 1. The feature vector is a histogram where each bin represents the frequency of each codeword that can be found for signal $s$. The rationalization behind this representation is

**Algorithm 2:** Encoding Stage

```
Input: s: a signal accelerometer sample; L: length of
       sliding window; O: number of timestamps that
       overlap between consecutive windows; B: the
       codebook
Result: A feature vector F of the signal s.
F ← initialize vector of size |B| // Output feature vector
W ← {} // Segments extracted from s
W ← SplitInWindows(s,L,O)
foreach w in W do
  | i ← Index in B of codeword closest to w
  | F[i] ← F[i]++ //Increase count of codeword i
end
return F
```

TABLE II
HYPER-PARAMETERS FOR THE CLASSIFIERS USED IN OUR EXPERIMENTS

| Classifier | Hyper-parameters |
|---|---|
| MLP | One hidden layer with 100 neurons, ReLU as activation function. |
| RF | 300 estimators |
| KNN | $K = 5$ |
| GNB | No parameters required |

that a sample signal belonging to a class $C_i$ will share the distribution of codewords of the other signals in the same class.

### C. Classifiers and Implementation

Based on the proposal of Júnior *et al.* [14], and motivated by the book by Domingos [31], we selected four classifiers that cover the major philosophies in the Machine Learning arena: a Multilayer perceptron Neural Network (MLP), a Random Forest (RF), a Naïve Bayes Classifier (GNB), and a K-nearest Neighbor algorithm (KNN). The implementation of these classifiers is the one offered by the scikit-learn 0.19.1 Machine Learning library [32]. The other related code (redundant signal removal, feature extraction with the Bag of Words model, and classifier evaluation) were implemented with Python 2.7 and can be found in the same repository used for data.

A grid search to tune the main parameters of these four classifiers was performed, choosing those that achieved the best classification scores on a sample of the data (the data used for this was not considered for the testing phase in the experimental section). These final parameters are presented in Table II.

### D. Evaluation Metrics

Three metrics were employed in order to evaluate the performance of the different methodologies, defined in terms of the True Positives (TP), True Negatives (TN), False Positives (FP), and the Total number of observations (T). Accuracy (Acc.) is defined as

$$Accuracy = \frac{TP + TN}{T} \quad (1)$$

The popular F-measure (F, of F1), which harmoniously considers both precision and recall, is defined in Equation 2:

$$F = \frac{2TP}{(2TP + FP + FN)} \quad (2)$$

To complement the F-measure, we also calculate the G-means [33], which is an appropriate score for unbalanced data sets, because it penalizes results when the classifier is better at correctly identifying positive over negative examples, and viceversa. We calculated G-means (G) as specified in Equation 3.

$$G = \sqrt{\frac{TP}{(TP + FN)} \times \frac{TN}{(TN + FP)}} \quad (3)$$

For all experiments we report the average results for twenty stratified shuffle splits, with 80% of data assigned for training and 20% for testing.

## V. EXPERIMENTS FOR AGGRESSIVE DRIVING BEHAVIOR DETECTION

In this section we present results with respect to three experiments. In the first experiment we focus on evaluating the ability of the classifiers and data representations to perform binary classification, i.e., given an acceleration sample to determine if it corresponds to either an example of aggressive or safe driving behavior. To contrast our findings, we compare our results with the ones reported in [17].

For the second experiment we go one step further, since we now focus in measuring the ability of the approaches to tackle a multi-class classification problem, this is, correctly assigning a class-label to a specific aggressive driving maneuver, e.g. *was it an abrupt swerve, a sudden brake or a non-aggressive event?* Accordingly, we contrast the machinery reported by Ferreira *et al.* [14] against our own proposal, both in theirs and in our own dataset.

Finally, in the third experiment, we look to gain some insight about how the BoW approach compares against successful approaches based on some variations of the same idea, such as Bag of Patterns [18], and a Bag of Words over a Symbolic Aggregate Approximation (SAX) [19].

### A. Experiment 1: Binary Classification – Discriminating Aggressive From Safe Driving Behavior

Žylius recently approached this experiment, creating an original data set and testing different combinations of features to compose a discriminative vector, with the best accuracy score being 95.5% [17]. Up to 78 different features were derived in his work, both in the time and frequency domain, so by tackling this first experiment we are comparing our proposal against a very exhaustive study of features. For this experiment, we perform a grid search of parameters for the BoW model that directly influence its ability to represent patterns in accelerometer data. Table III presents the parameters that we evaluated. Note that we do not only consider different

TABLE III
PARAMETERS CONSIDERED TO TUNE THE BOW

| Parameter | Evaluated value |
|---|---|
| Accelerometer axes | X, Y, $\|(X,Y)\|_2$, XY |
| Classifiers | MLP, RF, GNB, KNN |
| Codebook size | $K = \{2, 5, 10, 20, 50, 100, 150, 200, 250\}$ |
| Event's length (time stamps) | $L = \{5, 10, 20, 50, 100, 150, 200, 250, 300\}$ |
| Overlapping (50%) | Yes/No |
| TF-IDF | Yes/No |

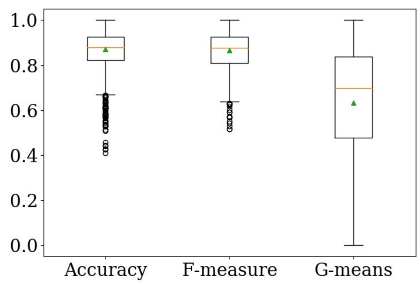



Fig. 4. Box plots summarizing the distribution of classification results when using BoW.

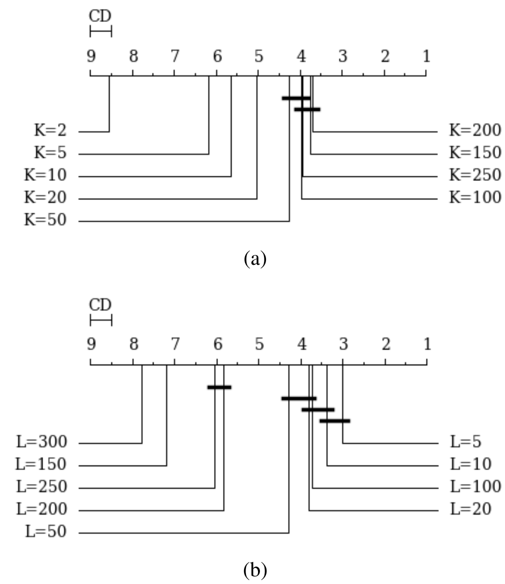



Fig. 5. (a) CDD for F-measure with respect to $K$ (codebook size). (b) CDD for F-measure with respect to $L$ (time stamps). Acceleration data was captured at 17 Hz.

classifiers and BoW parameters, but also accelerometer axes[6] (or a fusion of them via concatenation, or a norm), and the usage of a factor called Term frequency – Inverse document frequency weighting (TF-IDF) [34], commonly used to add discriminative power to the Bag of Words model in Natural Language Processing (NLP) tasks.

Out of the evaluation of 5,184 parameter combinations, we found that 554 (10.6% of the total evaluated) yielded an accuracy of at least 0.955, that is, as good as or better than the one reported in [17]. Figure 4 shows the distribution of the three evaluated metrics for all the combinations of parameters. This figure supports our logic of computing the G-means score, since there are some configurations that yield a high accuracy but with a very low G-means, possibly leading to erroneous conclusions if only accuracy were to be reported.[7]

Table IV presents the top five performers on the three metrics, interestingly none of them misclassified any driving event. Given the many combinations of parameters that we explored, further analysis is required to produce a recommendation on which values are the best to use. Therefore, Kruskal-Wallis H-tests and post hoc Nemenyi tests, with a level of significance $\alpha = 0.05$, were applied to the results obtained with the individual parameters of the model. To show this analysis we use the Critical Difference Diagram (CDD) [35]. These diagrams show in an elegant manner how different configurations are ranked (the rightmost having the best result), indicating at the same time (joined with a thick bar) when two or more configurations have no statistical difference. From this analysis we make the following generalizations:

- The usage of the norm of the vectors for longitudinal and lateral acceleration yields better results than the other evaluated sources of information. This was expected since longitudinal and lateral accelerations tend to be used to detect different types of driving events (braking/cornering, etc.). Using only the axis associated to Lateral acceleration is the second best alternative.
- Larger codebooks ($K > 50$) tend to perform statistically better than smaller ones. See Figure 5a.
- In general, smaller event lengths $L$ achieve better results. See Figure 5b. This indicates that it is sufficient to consider driving events with a temporal length of about 6 seconds when aiming to differentiate between aggressive and safe driving.
- Overall, MLP was the best option to classify the feature vectors computed with BoW, followed by a GNB, but there's no significant difference between both classifiers in terms of G-means. However, the GNB classifier had a much lower correlation between F-measure and G-means than the MLP (0.76 and 0.96, respectively), so MLP is preferred.

### *B. Experiment 2: Multi-Class Classification of Aggressive Driving Maneuvers*

For this experiment we are interested in addressing the classification of events into aggressive driving maneuvers such as swerving, sudden braking or sudden acceleration, among others. The importance of this task relies on the fact that a correct classification of a risky maneuver could translate to a faithful driving score calculation. Note that this time we limited the word size to 200 timestamps (the maximum length of the padded driving events). Figure 6 summarizes the

[6]Since the vertical axis is perpendicular to the road and almost no acceleration related to a driving maneuver affects it, this axis was not considered in the evaluation. This convention is well-established in the literature where accelerations in the longitudinal direction are used to detect harsh forward acceleration and braking events, whereas accelerations in the lateral direction pick up swerving, cornering, and lane-changing [13].

[7]This can be the case when one class is extremely large. If the classifier predicts that all examples belong to that class, the accuracy will be high; however, G-means penalizes this case, yielding a very low score.

TABLE IV

RESULTS OF THE TOP FIVE BOW CONFIGURATIONS IN ZYLIUS' DATASET [17]

| Classifier | K | L | Overlap | TF-IDF | F-measure | G-means | Accuracy |
|---|---|---|---|---|---|---|---|
| MLP | 250 | 100 | No | Yes | 1 | 1 | 1 |
| MLP | 150 | 100 | Yes | Yes | 1 | 1 | 1 |
| MLP | 150 | 50 | No | Yes | 1 | 1 | 1 |
| MLP | 100 | 100 | No | Yes | 1 | 1 | 1 |
| MLP | 50 | 50 | No | Yes | 1 | 1 | 1 |

TABLE V

BEST RESULTS FOR THE MULTI-CLASS CLASSIFICATION PROBLEM OVER THE AUTHOR'S DATASET

| Methodology | Classifier | nf | K | L | Overlap | TF-IDF | F-measure | G-means | Acc. |
|---|---|---|---|---|---|---|---|---|---|
| Authors' | GNB | - | 150 | 200 | Yes | No | 0.9702 | 0.9758 | 0.9688 |
| Authors' | MLP | - | 50 | 150 | Yes | No | 0.9689 | 0.9807 | 0.9683 |
| Authors' | MLP | - | 150 | 200 | Yes | Yes | 0.9689 | 0.9828 | 0.9677 |
| Authors' | MLP | - | 100 | 200 | No | Yes | 0.9669 | 0.9679 | 0.9661 |
| Authors' | MLP | - | 100 | 150 | Yes | Yes | 0.9665 | 0.9776 | 0.9655 |
| [14] | RF | 4 | - | - | - | - | 0.8738 | 0.9244 | 0.8725 |
| [14] | RF | 3 | - | - | - | - | 0.8370 | 0.8949 | 0.8351 |
| [14] | RF | 2 | - | - | - | - | 0.6560 | 0.7701 | 0.6586 |
| [14] | RF | 1 | - | - | - | - | 0.4662 | 0.6099 | 0.4781 |

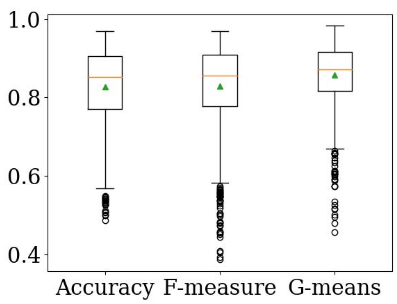



Fig. 6. Box plots summarizing the distribution of the average results for individual maneuver classification.

distribution of these results when BoW is employed. As can be seen, BoW is robust and consistently offers competitive results in the three metrics.

Now, we evaluate the methodology reported in [14]. In their work, they presented a feature vector based on statistical summarization of *frames* (segments with different length). Unlike we do in this study, they process sliding windows of long time series (mostly containing normal driving, and few examples of aggressive maneuvers) obtained from an array of different sensors and considering several categories of events. When we evaluate their methodology in our data set, we transform the acceleration samples to a feature vector that considers one to four frames ($nf$) with length of 50 timestamps (about one second), and use Random Forest (the same classifier recommended in their article).

Table V contrasts the best results obtained with BoW against those of Ferreira *et al.*'s machinery applied over our data set. We observe that with Ferreira *et al.*'s feature vector a RF is capable of classifying different types of maneuvers with the usage of multiple frames of acceleration readings per event. However, BoW models clearly outperform it in all three of the evaluated metrics.

After performing the same statistical tests as in the previous section it was found that, in general, the best results were obtained with $L = \{100, 150, 200\}$ (two to four seconds of signal, the complete duration of most events) and high $K$ values ($K = \{20, 50\}$).[8] The usage of TF-IDF tends to improve results in terms of F-measure. Again, MLP generally achieves the higher performance, with GNB and RF yielding competitive results.

*1) On Ferreira et al.'s Data Set:* We performed the same evaluation on the data set provided by Júnior *et al.* [14]. Table VI condenses best results of both approaches in their data set. Although results suggest that both approaches found this data set harder to classify (in comparison with the one proposed in this study), the BoW approach continues outperforming Ferreira *et al.*'s proposal in all the evaluation scores. Remarkably, even when this data set has less than one fifth of the examples in our data set and almost twice the number of classes, BoW shows a robust performance.

### *C. Experiment 3: Evaluation of Other Variants of the Bag of Words Approach*

So far, results suggest that the Bag of Words representation is a robust and effective way to detect aggressive driving patterns in accelerometer data. However, since there are different flavors of Bag of Words that could be used, based on the comparison of tree different ideas we aim to declare what is the best option to use. First, we explore the usage of SAX [19], [36] for the conversion of time series to a text representation, on which we apply a Bag of *n-grams*. After this,

[8]Note that this finding agrees with what we previously reported in experiment 1, that accelerometer signal of about 6 seconds should be enough to discriminate between aggressive and safe driving. That is, for a binary case the accelerometer sampling rate was 17 Hz, so 6 seconds is 102 time stamps, which is within the range of L that we found to offer the best results for this experiment.

TABLE VI
BEST RESULTS FOR THE MULTI-CLASS CLASSIFICATION PROBLEM OVER FERREIRA *et al.*'S DATASET [14]

| Methodology | Classifier | nf | K | L | Overlapping | TF-IDF | F-measure | G-means | Acc. |
|---|---|---|---|---|---|---|---|---|---|
| Authors' | RF | - | 10 | 5 | Yes | No | 0.8987 | 0.9184 | 0.8928 |
| Authors' | MLP | - | 5 | 100 | Yes | No | 0.8806 | 0.8967 | 0.875 |
| Authors' | RF | - | 50 | 5 | Yes | Yes | 0.8787 | 0.9148 | 0.8785 |
| Authors' | RF | - | 10 | 150 | Yes | Yes | 0.8736 | 0.8960 | 0.8714 |
| Authors' | MLP | - | 50 | 10 | Yes | Yes | 0.8601 | 0.9266 | 0.8607 |
| [14] | RF | 4 | - | - | - | - | 0.6862 | 0.8344 | 0.7013 |
| [14] | RF | 3 | - | - | - | - | 0.6381 | 0.8044 | 0.6473 |
| [14] | RF | 2 | - | - | - | - | 0.4938 | 0.7181 | 0.5200 |
| [14] | RF | 1 | - | - | - | - | 0.4564 | 0.6056 | 0.4709 |

TABLE VII
PARAMETERS CONSIDERED TO TUNE SAX-BASED CLASSIFICATION

| Parameter | Evaluated values |
|---|---|
| Accelerometer axes | XY |
| Classifiers | MLP, RF, GNB, KNN |
| Word length | $L = \{25, 50, 100, 150, 200\}$ |
| Alphabet length | $S = \{5, 10, 15, 20\}$ |
| N-gram max. size | $N = \{1, 2, 3, 4, 5\}$ |
| TF-IDF | Yes/No |

TABLE VIII
PARAMETERS CONSIDERED TO TUNE BOF-BASED CLASSIFICATION

| Parameter | Evaluated value |
|---|---|
| Accelerometer axes | XY |
| Classifiers | MLP, RF, GNB, KNN |
| Number of sequences | Default |
| Subsequence length factor | $z = \{0.1, 0.25, 0.5, 0.75\}$ |
| Histogram bins | $B = \{5, 10\}$ |
| Subsequence generation scheme | Totally random |
| TF-IDF | Yes/No |

we then turn to a Bag of Features [18], a representation based on the creation of histograms over statistical summarization of random segments. The evaluation of these BoW flavors will be carried out in the two data sets that we used for the multi-class classification problem.

*1) SAX-Based Bag of Words:* Time series converted to SAX are piecewise aggregate approximations represented by a series of letters [19]. A time series becomes a string, with length $L$, constructed from an alphabet of $S$ symbols. In order to perform this conversion, the time series is first standardized to zero mean and unit standard deviation, and is then down-sampled by averaging subsequences of timestamps (if $L$ is less than the length of the original series). $S$ determines the granularity of the piecewise approximation of acceleration values, that is, it indicates how many possible discrete values are used to represent the value of each time stamp (or averaged for a subsequence, if down-sampling).

This strategy was already used in [36] with the aim to classify aggressive driving maneuvers, so it would be interesting to see how these ideas actually compare against regular BoW. In order to apply a Bag of Words model we first used saxpy[9] to transform each padded time series to a SAX representation, and then segment the resulting strings in n-grams with lengths from 1 to $N$ characters. Histograms of these n-grams are calculated for the lateral (X) and longitudinal (Y) axes, and the concatenation of both is fed to a classifier. To find the best configuration of this strategy a grid search was conducted with the parameters presented in Table VII.

*2) Bag of Features:* Bag of Features (BoF) [18] is a framework proposed to classify Time Series. To the best of our knowledge, BoF has not been employed to represent accelerometer data in Driving Analytics applications, so this will serve to gain some insight about its potential application to this type of problems. Specifically, it extracts features by statistically summarizing randomly selected subsequences that differ in temporal location and length, these local features are then considered as an independent data set (with labels from their original time series). A Random Forest classifier is used to obtain class probability estimates. Histograms of these probability estimates are concatenated and global features are added, the resulting vector is considered a representation of the original time series.

We processed the data set with the implementation kindly provided by the authors of BoF,[10] concatenating the BoF representation vectors for axes X and Y before performing classification. The parameters that we considered to tune this approach are the ones presented in Table VIII.

*3) Results of the Comparison of Variants of BoW:* Table IX presents F-measure score for each of the aggressive maneuvers contained in the Authors' and Ferreira *et al.*'s data sets, respectively. According to this, we can observe that all three bag of words models manage to produce very competitive classification scores when representing accelerometer data. This result suggests that a BoW-based strategy must be considered in further comparisons with other strategies, and perhaps even as a baseline. The BoW approach model achieved the best result, followed by the Bag of Features representation.

Interestingly, even if their underlying principles are different, it was observed that all three representations produce the best output when longer sequences of timestamps are considered to extract features. This is probably related to the low frequency changes associated with the events being more representative than their high frequency components.

[9]https://github.com/nphoff/saxpy

[10]http://www.mustafabaydogan.com/a-bag-of-features-framework-to-classify-time-series-tsbf.html

TABLE IX
F-SCORE BY EVENT TYPE BY THE BEST CLASSIFIERS

| **Event** | BoW | BoF | SAX |
|---|---|---|---|
| Sudden acceleration | 0.98 | 0.97 | 0.94 |
| Sudden braking | 0.98 | 0.98 | 0.95 |
| Swerve left | 0.93 | 0.93 | 0.91 |
| Swerve right | 0.95 | 0.93 | 0.89 |
| Average | **0.96** | 0.95 | 0.92 |
| Aggressive acceleration | 0.91 | 0.76 | 0.78 |
| Aggressive braking | 0.98 | 0.96 | 0.94 |
| Aggressive left lane change | 0.77 | 0.69 | 0.59 |
| Aggressive left turn | 0.79 | 0.81 | 0.82 |
| Aggressive right lane change | 0.96 | 0.83 | 0.76 |
| Aggressive right turn | 0.86 | 0.79 | 0.69 |
| Non-aggressive event | 0.93 | 0.66 | 0.53 |
| Average | **0.88** | 0.78 | 0.73 |

A hypothesis that could explain the outstanding performance of the BoW representation is that the clustering part of the process could be finding *shapelets* [37], subsequences found in time series that are highly representative of a class. This could also apply to SAX, however, the poorer performance of this latter representation might be explained by the loss in granularity produced by the Piecewise Aggregate Approximation step on the short time series that we analyzed. On its part, the Bag of Features representation, consistently outperforms SAX, falling behind BoW just barely in the authors' data set.

## VI. CONCLUSION

Unsupervised automatic feature extraction has seen very little use in the context of acceleration-based driving style detection. In this paper, we explored the usage of Bag of Words (BoW) models for aggressive driving detection. Particularly, we approached two problems. The first problem was a binary classification problem where the goal is to differentiate between aggressive and safe driving behavior. For this problem the BoW model clearly outperformed a recent work [17], yielding a perfect score (zero misclassification) on the same data set. For the second problem, the goal was to assign a class-label out of several to name the particular type of aggressive maneuver example. In this part, we compared BoW against the state-of-the-art work reported in Júnior *et al.* [14], outperforming the latter with a difference of 8% in G-means. Moreover, we proposed our own data set for this multi-class classification problem and compared again the two mentioned approaches, resulting in BoW being better than theirs by 5%. Motivated by these results, we also tested other models that are also based on Bag of Words, such as Bag of Features and SAX. In this comparison, BoW was superior over the second best strategy with a difference in F-measure of 1% in our own data set, and by 10% in Ferreira *et al.*'s data set. In summary, the presented results demonstrate that the Bag of Words model positions itself as a highly competitive method to represent accelerometer data with the aim to boost driver behavior classification. Hence, BoW should be able to quickly find use within intelligent transportation applications such as insurance telematics and general driving style recognition. As future work we would like to evaluate input from some other sensors such as Gyroscopes and Magnetometers, and see if the BoW model is able to find an effective representation for data of a different nature.

## REFERENCES

[1] K. H. Beck, "Driving while under the influence of alcohol: Relationship to attitudes and beliefs in a college population," *Amer. J. Drug Alcohol Abuse*, vol. 8, no. 3, pp. 377–388, Jan. 1981.
[2] D. J. Beirness, "Do we really drive as we live? The role of personality factors in road crashes," *Alcohol Drugs Driving*, vol. 9, no. 4, pp. 129–143, 1993.
[3] J. Arnett, "Drunk driving, sensation seeking, and egocentrism among adolescents," *Personality Individual Difference*, vol. 11, no. 6, pp. 541–546, 1990.
[4] P. Händel *et al.*, "Insurance telematics: Opportunities and challenges with the smartphone solution," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 4, pp. 57–70, Oct. 2014.
[5] J. Engelbrecht, M. J. Booysen, G.-J. van Rooyen, and F. J. Bruwer, "Survey of smartphone-based sensing in vehicles for intelligent transportation system applications," *IET Intell. Transp. Syst.*, vol. 9, no. 10, pp. 924–935, Dec. 2015.
[6] G. A. M. 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, Dec. 2015.
[7] E. I. Vlahogianni and E. N. Barmpounakis, "Driving analytics using smartphones: Algorithms, comparisons and challenges," *Transp. Res. C, Emerg. Technol.*, vol. 79, pp. 196–206, Jun. 2017.
[8] D. A. Johnson and M. M. Trivedi, "Driving style recognition using a smartphone as a sensor platform," in *Proc. 14th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2011, pp. 1609–1615.
[9] M. Fazeen, B. Gozick, R. Dantu, M. Bhukhiya, and M. C. González, "Safe driving using mobile phones," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1462–1468, Sep. 2012.
[10] J. Dai, J. Teng, X. Bai, Z. Shen, and D. Xuan, "Mobile phone based drunk driving detection," in *Proc. 4th Int. Conf. Pervasive Comput. Technol. Healthcare*, Mar. 2010, pp. 1–8.
[11] H. Eren, S. Makinist, E. Akin, and A. Yilmaz, "Estimating driving behavior by a smartphone," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 234–239.
[12] L. C. González, R. Moreno, H. J. Escalante, F. Martínez, and M. R. Carlos, "Learning roadway surface disruption patterns using the bag of words representation," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 11, pp. 2916–2928, Nov. 2017.
[13] J. Wahlström, I. Skog, and P. Händel, "Smartphone-based vehicle telematics: A ten-year anniversary," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 10, pp. 2802–2825, Oct. 2017.
[14] J. F. Júnior, E. Carvalho, B. V. Ferreira, C. de Souza, Y. Suhara, A. Pentland, and G. Pessin, "Driver behavior profiling: An investigation with different smartphone sensors and machine learning," *PLoS One*, vol. 12, no. 4, Apr. 2017, Art. no. e0174959.
[15] M. E. Aragón, M. R. Carlos, L. C. G. Gurrola, and H. J. Escalante, "A machine learning pipeline to automatically identify and classify roadway surface disruptions," in *Proc. 16th Mex. Int. Conf. Comput. Sci.*, Nov. 2016, p. 3.
[16] Y. Goldberg, "Neural network methods for natural language processing," *Synth. Lectures Hum. Lang. Technol.*, vol. 10, no. 1, pp. 1–309, 2017. doi: 10.2200/S00762ED1V01Y201703HLT037.
[17] G. Žylius, "Investigation of route-independent aggressive and safe driving features obtained from accelerometer signals," *IEEE Intell. Transp. Syst. Mag.*, vol. 9, no. 2, pp. 103–113, Apr. 2017.
[18] M. G. Baydogan, G. Runger, and E. Tuv, "A bag-of-features framework to classify time series," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2796–2802, Nov. 2013.
[19] P. Senin and S. Malinchik, "SAX-VSM: Interpretable time series classification using SAX and vector space model," in *Proc. IEEE 13th Int. Conf. Data Mining*, Dec. 2013, pp. 1175–1180.
[20] L. Eboli, G. Mazzulla, and G. Pungillo, "Combining speed and acceleration to define car users' safe or unsafe driving behaviour," *Transp. Res. Part C, Emerg. Technol.*, vol. 68, pp. 113–125, Jul. 2016.
[21] L. Eboli, G. Mazzulla, and G. Pungillo, "How to define the accident risk level of car drivers by combining objective and subjective measures of driving style," *Transp. Res. Part F, Traffic Psychol. Behav.*, vol. 49, pp. 29–38, Aug. 2017.

[22] M. R. Carlos, M. E. Aragón, L. C. González, H. J. Escalante, and F. Martínez, "Evaluation of detection approaches for road anomalies based on accelerometer readings—Addressing Who's Who," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 10, pp. 3334–3343, Oct. 2018.

[23] B. Predic and D. Stojanovic, "Enhancing driver situational awareness through crowd intelligence," *Expert Syst. Appl.*, vol. 42, no. 11, pp. 4892–4909, Jul. 2015.

[24] M. M. Bejani and M. Ghatee, "A context aware system for driving style evaluation by an ensemble learning on smartphone sensors data," *Transp. Res. C, Emerg. Technol.*, vol. 89, pp. 303–320, Apr. 2018.

[25] J. Engelbrecht, M. J. Booysen, G.-J. van Rooyen, and F. J. Bruwer, "Performance comparison of dynamic time warping (DTW) and a maximum likelihood (ML) classifier in measuring driver behavior with smartphones," in *Proc. IEEE Symp. Comput. Intell.*, Dec. 2015, pp. 427–433.

[26] M. R. Carlos, L. C. González, F. Martínez, and R. Cornejo, "Evaluating reorientation strategies for accelerometer data from smartphones for ITS applications," in *Proc. Int. Conf. Ubiquitous Comput. Ambient Intell.*, Nov. 2016, pp. 407–418.

[27] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. Hoboken, NJ, USA: Wiley, 2012.

[28] R. Larsdotter and D. Jaller, "Automatic calibration and virtual alignment of MEMS-sensor placed in vehicle for use in road condition determination system," M.S. thesis, Chalmers Sweden, SK, USA, Univ. Technol., Jan. 2014.

[29] J. Wang, P. Liu, M. F. H. She, S. Nahavandi, and A. Kouzani, "Bag-of-words representation for biomedical time series classification," *Biomed. Signal Process. Control*, vol. 8, no. 6, pp. 634–644, 2013.

[30] J. Lin and Y. Li, "Finding structural similarity in time series data using bag-of-patterns representation," in *Proc. Int. Conf. Sci. Stat. Database Manage.*, Jun. 2009, pp. 461–477.

[31] P. Domingos, *The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World*. 1st ed. New York, NY, USA: Basic Books, 2015.

[32] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[33] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-sided selection," in *Proc. (ICML)*, Jul. 1997, pp. 179–186.

[34] G. Salton and M. McGill, *Introduction to Modern Information Retrieval*, 1st ed. New York, NY, USA: McGraw-Hill, 1983.

[35] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.

[36] P. Chaovalit, C. Saiprasert, and T. Pholprasit, "A method for driving event detection using SAX with resource usage exploration on smartphone platform," *J. Wireless Commun. Netw.*, vol. 135, pp. 1–11, Dec. 2014.

[37] L. Ye and E. Keogh, "Time series shapelets: A new primitive for data mining," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Disc. Data Mining*, Jun. 2009, pp. 947–956.

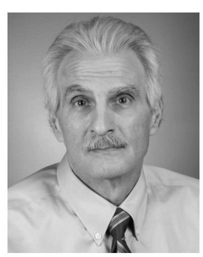
**Manuel Ricardo Carlos** received the B.Eng. degree in computer systems' engineering from the Instituto Tecnológico de Chihuahua II (ITCH II) in 2006, and the M.Eng. degree in software engineering from the Universidad Autónoma de Chihuahua (UACH) in 2015, where is currently pursuing the Ph.D. degree in engineering.

**Luis C. González** received the Ph.D. degree from the University of North Carolina at Charlotte in 2011. He is currently a member of the National System of Researchers, National Council of Science and Technology of México (CONACyT). He is also with the Facultad de Ingeniería, Universidad Autónoma de Chihuahua, Mexico. His current research interests include pattern recognition and combinatorial optimization problems. In 2017, he received the Latin America Google Research Award for his work on intelligent transportation systems.

**Johan Wahlström** received the B.Sc., M.Sc., and Ph.D. degrees from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2013, 2014, and 2017, respectively. He has been a Post-Doctoral Researcher with Oxford University since 2018. In 2015, he spent one month at the University of Porto and six months at Washington University in St. Louis as a Visiting Ph.D. Student. He was accepted into the program of excellence in electrical engineering at KTH in 2014, and in 2015, he was the youngest recipient of the Sweden-America Foundation's Research Scholarship.

**Graciela Ramírez** received the M.Sc. and Ph.D. degrees in electrical engineering from the Instituto Tecnológico de Chihuahua, Mexico. From 2005 to 2012, she was with Visteon Automotive Systems as an Electrical Engineer. She is currently a Professor with the Universidad Autónoma de Chihuahua and a member of the National System of Researchers, National Council of Science and Technology of Mexico.

**Fernando Martínez** received the master's degree in industrial electronic engineering from the Instituto Tecnológico de Chihuahua, Chihuahua, México, in 1997, and the Ph.D. degree in computer science from the University of Nottingham, Nottingham, U.K., in 2009. He is currently with the Facultad de Ingeniería, Universidad Autónoma de Chihuahua, Mexico. His current research interests include mobile computing, sensor networks, and human–computer interactions.



**George Runger** received the Ph.D. degree in statistics from the University of Minnesota in 1982. He was a Senior Engineer and a Technical Leader for data analytics projects with IBM. He is currently a Professor with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ, USA. He has authored over 100 publications in research journals with funding from federal and corporate sponsors.