Stop Purpose Classification from GPS Data of Commercial Vehicle Fleets

Leonardo Sarti, Luca Bravi, Francesco Sambo, Leonardo Taccari,
Matteo Simoncini, Samuele Salti and Alessandro Lori
Fleetmatics Research
Via Paisiello 20, Florence, Italy
Email: luca.bravi@fleetmatics.com

***Abstract*—Extracting stop purpose information from raw GPS data is a crucial task in most location-aware applications. With the continuous growth of GPS data collected from mobile devices, this task is becoming more and more interesting; a lot of recent research has focused on pedestrians (mobile phones) data, while the commercial vehicles sector is almost unexplored.**

In this paper we target the problem of stop identification and classification from vehicle GPS data, using a large and heterogeneous dataset of commercial fleets from diverse industries. Our aim is to classify stops by purpose in two categories: work related and non-work related.

Our dataset consists of more than 700k stops, 160k of which are work stops. For each stop, we compute a set of 100 different features, which can be grouped in 4 main categories: stop-wise features, points of interest features, stop cluster features, and sequential features. By choosing Random Forests as classification model, we are able to assess the relative importance of each of the features in the four sets.

Experimental results show that our method significantly outperforms the state of the art models for stop purpose classification in the context of commercial vehicles. The feature ranking highlights the importance, for classifying a stop, of both its duration and the duration of other stops in the same location.

I. Introduction

In the last decade, the great diffusion of GPS (Global Positioning System) devices is generating a growing interest in the application of data mining algorithms to the huge amount of spatio-temporal data produced by such devices. Two examples are the travel mode detection problem [1]–[4] or the vehicle classification problem [5], [6].

Many practical applications require the use of semantic information about user behaviors and geographical locations; two different examples are [7], where semantic information is used to measure similarities between users based on their location history, and [8], where a location-based recommender for interesting places is proposed.

Of particular interest, then, are tasks that involve semantic tagging of GPS data: among them, semantic place detection and stop or trip purpose identification have the aim of identifying and classifying locations, along GPS trajectories, that are significant to the users (one, or all of them). It is worth noting that these problems, though not equivalent, are strongly related and, in a sense, complementary: for example, co-located stops that have the same purpose for multiple users likely correspond to semantically relevant places; and vice versa, knowing the semantics of a place can be of great help in classifying the purpose of a single stop.

Methods for such problems typically involve two phases: *detection* of interesting places and *classification* of the detected locations. Place detection is typically tackled in an unsupervised fashion, as in [9] and [10], where the authors propose clustering methods based on a modified version of the DBSCAN algorithm [11], or [12], where hierarchical clustering is exploited to extract visit points from stop locations. When computational performance is an issue, e.g. for very large datasets, other works exploit simpler but more scalable approaches that rely on a hashing of the place coordinates [13], [14]. Concerning place and/or stop classification, two main approaches can be found in the literature: rule-based systems [15], [16], that rely mostly on the position of the activity and on land use data, and machine learning approaches, that focus more on characteristics extracted from the activity itself. An example of the latter approach is [10], where an SVM classifier is used to discriminate between activity and non-activity stops among the identified places. The three major features extracted for the SVM method are the stop duration, the mean distance to the centroid of the points surrounding each stop location, and the shortest among the distances from the current location to home and to the workplace. In [12], temporal and spatial features are exploited by means of a classifier (SVM, random forest or logistic regression) and sequential features by means of a Hidden Markov Model (HMM) to categorize them into predefined types. In [14], semantic classification of places is performed based on a combination of GPS data and satellite images.

In this article, we are mainly concerned about a specific version of the stop purpose classification problem that considers the classification of stops of commercial vehicle fleets. In this context, the problem typically consists in categorizing stops into service stops (*work orders*) and personal or non-work stops. Some examples of the latter are resting stops, refueling, maintenance and night stops. The problem is extremely relevant for fleet intelligence companies: having a correct automatic classification of work and non-work stops allows for much richer information to be provided to users (e.g., fleet managers).

Only a limited number of recent papers address this problem. An example is [17], where a Support Vector Machine

(SVM) is applied to identify delivery stops using GPS data. Stop duration, the distance to the center of the city, and the presence of a bottleneck (such as a bridge, a toll booth, or a tunnel) in the neighborhood of the stop are the three features used in the SVM model. A case study using second-by-second GPS data in New York City shows high accuracy results, although the amount and the variety of data used is rather limited.

A broader dataset is used in [13]: in order to determine the purpose of stopped truck events, the concept of entropy is applied to a large volume of unlabeled GPS data, composed of about 100 millions of data pings coming from 40 thousand vehicles and a few hundred fleets across Canada in March 2013. The authors categorize stop events into two types: primary stops, where goods are transferred, and secondary stops, where vehicle and driver needs are met, such as rest stations. The proposed entropy technique measures the diversity of the truck fleets with trucks that dwell for 15 minutes or longer at a given location. They show that secondary stops usually exhibit larger entropy, that arises from a greater variety of fleets and an even distribution of stop events among these fleets, while, conversely, primary shipping depots and other locations where goods are transferred tend to have lower entropy, due to the lower variety of fleets that exploit such locations. The authors validated the 150 locations with the highest entropy by looking at Google Maps and Google Street View to determine the type of stop, resulting in 148 out of the 150 identified clusters actually being in correspondence with truck stops, gas stations, and several motels.

In this paper, we address the same problem of [17] and [13], stop purpose classification in commercial fleets. We employ a random forest classifier exploiting four different sets of features. The data we use comes from the customers of one of the leading companies in the field of vehicle tracking system. The main contributions of this paper are the following:

- we describe a method to extract stops from GPS pings and to assign them their ground truth label from work stop schedule information;
- based upon the labeled dataset, we build a model that can automatically classify a stop between two classes: work stops and non-work stops;
- we provide and rigorously evaluate different sets of features to solve our problem, including:
  - stop-wise features;
  - points of interest (POI) features;
  - stop cluster features;
  - sequential features.

The remainder of this paper is structured as follows. Section II presents the methodology: in particular Section II-A provides a high-level description of the raw GPS and work stop data, Section II-B presents the algorithm to detect stops starting from raw GPS pings, Section II-C describes the procedure to associate ground-truth labels to the detected stops, and Section II-D introduces all the features that are used by our machine learning model. In Section III we present our experiments: Section III-A introduces the training procedure and the experimental settings, Section III-B describes the baseline methods, and Section III-C discusses the experimental results. Finally, Section IV reports conclusions and future directions.

TABLE I
TYPES OF COMPANIES IN THE DATASET.

Type	Count
Heating & air conditioning	45
Plumber & leak detection	18
Protection systems	9
Maintenance & cleaning	7
Electric	6
Irrigation & lighting	5
Delivery	4
Pools installation	2
Locksmith	1
Health assistance	1

II. METHODS

In this section, we first describe the structure of the GPS and work order data we use. Afterwards, we introduce the stop detection technique adopted to find groups of GPS pings that represent the same stops and how we assign them the ground truth labels. Finally, we describe the extraction of the features used by a random forest model to classify the stops either as ”Work Stop” or ”Non-work Stop”.

A. GPS and Work Order data

Our dataset was collected by Fleetmatics, a fleet intelligence company for 98 small and medium business (SMB) companies, i.e. with fewer than 100 vehicles, over one year of activity of vehicles in the USA (February 2015 - January 2016), resulting in more than 55 million GPS pings. In Table I we show the industry of the companies in the datasets. The dataset is, to the best of our knowledge, one of the the largest and most diverse among those used in the literature for similar problems.

The collected data is of two types: raw GPS pings, providing information on the position of the vehicles, and work order status messages, providing information on the schedule and progress of the jobs executed by the drivers.

A sequence of GPS pings $\{\mathbf{P}_i\}_{i=1}^n = \{\mathbf{P}_1, \ldots, \mathbf{P}_n\}$ characterizes the routes traveled by each vehicle. Each GPS ping $\mathbf{P}_i$ contains a vehicle id v_i, latitude and longitude (*i.e.*, position p_i), odometer distance d_i, a timestamp t_i, and and event code e_i, which gives us status information about the vehicle.

The typical sampling rate ranges from 1 to 2 minutes. In general, we cannot assume uniform sampling rates, as data collected by heterogeneous GPS devices may have different sampling rates, and the rate can vary in the same device due to the occurrence of asynchronous triggers, like *e.g.* harsh driving events, or *engine off* events, that automatically fire a GPS ping.

We can also lose GPS pings if the vehicle is in a zone not reached by the satellite.

The data about work orders, on the other hand, consists of sequences $\{\mathbf{W}_i\}_{i=1}^{n} = \{\mathbf{W}_1, \ldots, \mathbf{W}_n\}$, each containing a vehicle id v_i, latitude and longitude information (the position p_i), a timestamp t_i, and a status code c_i (for instance, *pending*, *started*, *completed*).

B. Vehicle stop detection

As explained in the previous paragraph, the raw data we use regarding the instantaneous vehicle positions is composed of GPS pings. However, we are interested in aggregating them to be able to describe the activity of the vehicles. To this end, we developed a spatio-temporal clustering procedure that first assigns a type to each GPS message and then gathers them into groups of GPS pings that we define as *stops*. GPS pings may belong to the following three classes:

- *engine off*: pings with an *engine off* event. These pings are generated the instant the engine is turned off (while the engine is off, no pings are sent);
- *idling*: pings where the engine is on, but the vehicle is still or moving slowly in a small area. Let $H(p_i, p_{i-1})$ be the haversine distance between two points. Then, given a pair of consecutive pings $\mathbf{P}_{i-1}$ and $\mathbf{P}_i$ of a given vehicle. we define them as *idling* if they satisfy the following constraints:
    1) $s_i = \frac{H(p_i, p_{i-1})}{t_i - t_{i-1}} \leq 1.4\, m/s$, to ensure that the speed is close to zero;
    2) $H(p_i, p_{i-1}) \leq 150\ m$, to ensure that $\mathbf{P}_i$ and $\mathbf{P}_{i-1}$ are actually close, and avoid artifacts due to lost messages.

  Both the thresholds have been selected after a preliminary qualitative analysis performed visually inspecting the obtained clusters;
- finally, *journey* pings are all those that are neither *engine off* nor *idling*.

After the pings have been categorized, they are sorted chronologically for each vehicle, and all the *idling* and *engine off* pings are assembled together, forming a group for all the consecutive pings which are not separated by *journey* pings. Due to missing data that may cause consecutive idling pings to be very far from each other, we enforce again the spatio-temporal constraints ($s_i \leq 1.4\, m/s$ and $H(p_i, p_{i-1}) \leq 150\, m$) within the group – when the constraints are not satisfied, the group is split into multiple ones. All *journey* pings are finally discarded.

The groups of pings created in this manner represent the identified vehicle stops that we want to classify. To sum up, a stop is defined as a group of chronologically consecutive pings which are either *idling* or *engine off* and satisfy further spatio-temporal locality constraints.

Each stop has several characteristics we can compute across the pings it includes: for instance, the number of aggregated pings, the start and end of the stop (the first and last timestamps of the GPS pings belonging to the stop), the stop duration (computed as the time between the end and the start of the stop), or its shape, defined by the max/min latitude/longitude coordinates of its GPS messages. All this information will be crucial to extract features in the second phase, that is, stop purpose classification.

Note that, according to our definition, a stop does not necessarily include an *engine off* ping, but may contain *idling* pings only. We do not discard stops with only *idling* pings because we have observed that they often represent work orders. For instance, drivers of delivery companies frequently make their deliveries without turning off the vehicle. Clearly these situations may lead to an ambiguity with the cases where the vehicle is stuck in the traffic or waiting at a traffic light.

We want to remark that our way of grouping close GPS pings into *stops* does not only lead to identifying places where the vehicles have been still, but also identifies areas where the vehicles had a certain kind of operations. A stop may include, for instance, the time spent parking a vehicle, but also the time spent in the close proximity, finding the parking place. Another example is the case where the vehicle stops and moves multiple times in a small area: this can happen, for instance, when a vehicle is at the depot and is loaded in multiple load points, or, similarly, when it is unloading his cargo to a client. This is why we preferred to use a speed threshold which is not too strict ($1.4\ m/s$, roughly $5\ km/h$) to assess that a vehicle stays still. A strict 0 speed threshold would also be hardly practical due to the GPS signal noise.

C. Labeling

In the second phase of our task, we aim to classify the purpose of each identified stop. To first obtain ground truth labels for the stops obtained with the above procedure, we match them with the work order data. In general, a stop of a vehicle is considered to be a work order if it matches both temporally and spatially a work order $\mathbf{W}_i$.

The spatial match is considered satisfied in a slightly different way for stops that only contain *idling* pings, and those with at least an *engine off*. The former (*idling*-only stops) are considered work order if their centroid is within an haversine distance of 150 meters with respect to the location of any entry of the work order dataset associated to the given vehicle. For stops with at least an *engine off* ping, the same condition must be valid for at least one *engine off* ping among those in the group. We treat *engine off* pings slightly differently because we believe their location is generally more meaningful than the others. Again the threshold has been chosen after a preliminary analysis of the percentage of the work orders matched with a stop which is not reported here for sake of readability.

Then, for any of the work orders spatially matching a stop, the temporal matching is satisfied if the time elapsed between the start and the end of the stop intersects the timespan between the work order entry indicating that the job has started and the one indicating that service has ended.

Since work order data is inserted by the fleet intelligence company's customers, we expect that a significant level of noise could exist in the database. Indeed, we cannot assume that all the customers used the same procedures to insert work order data in the database: for instance, work orders could be either inserted by the driver with a PDA, as soon as they are served, but they can be also inserted *a posteriori* by the fleet manager, leading to more uncertain data. In addition, the geocoding procedure is often affected by noise and the location of work orders might not be accurate.

Therefore, on the one hand, we do not want to be too strict in the matching between work orders and stops, lest we lose too many of them. On the other hand, we would also like to limit the amount of mislabeled data. We try to avoid the issue by not labeling ambiguous stops, that match spatially, but not temporally, a work order. Due to noise in the database, we believe that these cases might occur due to missing work order messages, or work orders with incorrect timestamps.

With the above mentioned stop labeling procedure, from our GPS and work order data we built a dataset composed of around 700k labeled entries, among which around 160k are work order stops, from 98 different vehicle fleets.

D. Classification features

Given the stops extracted along a sequence of GPS pings as described in Section II-B, we extract from them 100 different features that we use to train a Random Forest model. We divide the features into 4 different groups: stop-wise features (SWF), points of interest features (POIF), stop cluster features (CF) and sequential features (SeqF).

1) Stop-wise features: recalling that our stops come from the aggregation of multiple *idling* and/or *engine off* GPS pings, we define as stop-wise features:

- stop duration, computed as the time between the first and the last ping belonging to the stop;
- start time features: hour of day, day of week, day of month, day of year;
- time spent with the engine off: for each *engine off* event, we compute the time between that ping and the first following ping that is not an *engine off*, which represents the moment the engine is turned on. Since a stop can include multiple *engine off* pings, we aggregate the results using several aggregation functions (min, max, mean, variance, sum);
- shape: stop width, stop height, stop area, stop ratio, computed from the bounding box of the included GPS pings;
- stop type: *engine off* if it contains at least an *engine off* ping, *idling* otherwise;
- odometer distance from the first ping to the last ping of the stop;
- sum of haversines: sum of the pairwise haversine distances between consecutive GPS pings belonging to the stops;
- haversine distance between the first and the last ping of the stop (this feature, combined with the odometer and the stop shape features, help in distinguishing between stops where the driver moves around in a specific area, and the stops where the driver is moving straight, albeit very slowly);
- total count of pings in the stop;
- average speed (computed, as the odometer difference between the first and the last ping in the stop, divided by the duration of the stop);
- number of *engine off* pings in the stop.

Let us highlight that we use three different features measuring distance traveled inside the stop, because they are not always redundant. Indeed, we believe they help distinguish some cases, such as, for instance, when the distance is traveled in a queue, due to traffic congestion, and situations where a vehicle moves around in a confined area, e.g., while executing a work order in a courtyard. In the first case, the odometer distance and the haversine distance between the first and the last ping should be similar, while in the second one, the traveled odometer distance would likely be significantly bigger.

2) Points of interest features: We extend our initial pool of features by considering the presence of **points of interest (POI)** in the area surrounding each stop. To this end we use the cartographic service *PTV xLocate*[1] from which we extract the following POI types:

- bank;
- university;
- hotel;
- restaurant;
- rest area;
- grocery store;
- school;
- shopping center;
- fuel;
- open parking area;
- vehicle repair facility.

For each of these POI types, we build a feature that consists of the smallest distance of any POI of the given type from any of the *engine off* locations in the stop (or from the stop centroid, if there are none). If such distance is greater than 200 meters, an ∞ placeholder value is set to indicate the absence of nearby POIs of that type.

3) Stop cluster features: Another set of features, that we refer to as stop cluster features, is composed of some features that describe the characteristics of the stops surrounding the stop at hand. This way, we attempt to characterize the area of the stop: the rationale is that there are some areas where work orders and non work orders tend to cluster, in a similar way as what is proposed in [13] with an entropy measure. In particular, for each stop, we look inside a radius of 250 meters and collect the following statistics about the surrounding stops:

- vehicle entropy, computed as:

$$E_v = -\sum_{v \in V} \frac{n_v}{N} ln \left(\frac{n_v}{N} \right),$$

[1] http://xserver.ptvgroup.com/

where V is the set of vehicles of the given fleet, N is the total number of stops of all the vehicles of the fleet inside the radius of 250 meters, and n_v is the total number of stops of the vehicle v inside the same area. This entropy measure gives a sense of the variety, in an area, of vehicles of the same fleet;

- average, sum, max and min duration of the stops within the cluster;
- number of nearby stops (whose centroid is within 250 meters).

4) Sequential features: The work in [12] shows that taking into account the whole stop sequence of a user in a day is effective for the classification of personally semantic places. Even in a commercial context, one may think, for instance, that it is unlikely that a driver has two consecutive lunch stops. To this end we decided to include also some features carrying sequential information.

Considering the sequence of stops carried out by each vehicle, for each stop, we consider four neighboring stops: the two previous ones, and the two immediately following it. From these four stops, we compute a set of features based on their stop-wise and POI features, such as stop duration, time with engine off, distance from the closest restaurant, etc. The features to be extracted for each neighboring stop have been chosen in a preliminary investigation that we do not include here for sake of readability.

In addition, we compute:

- time from/to the previous/following stop;
- distance from/to the previous/following stop.

These features are useful, for instance, to identify stops that are far from others, meaning that the vehicle had to take a long detour to reach that place. This might be a hint that the place is semantically relevant.

Finally, among the sequential features, we also include two that are obtained from the *full* sequence of stops in the working day in which a stop occurred:

- ranking of the stop within the working day: on the date, we compute the (normalized) relative position where the stop occurred in the sequence of stops that the same vehicle carried out during the day (that is, the fraction of stops that occurred before the current stop on the same date);
- temporal percentage of the day covered: similar to the previous feature, but in this case it represents the fraction of time of the working day elapsed when the stop occurred. Note that this value is relative to the start of the working day, which is defined by the first *journey* ping of the day. This is an approximation based on the assumption that drivers rest during the night, which is the case for all the fleets in our dataset.

III. Experimental results

A. Training procedure

The labeled dataset of 702446 examples has been randomly split into training and test sets. To avoid bias between examples belonging to the same fleet, that are likely to have a similar behavior, the split is stratified by customer, in the sense that we keep all the entries of a given customer in only one of the two sets. The resulting training set is composed of 71 fleets, while the remaining 27 compose the test set. Detailed information on the cardinality of the two sets are showed in Table II.

The classification model we use is a Random Forest classifier [18], which consists of an ensemble of decision trees, and is widely believed to be among the best choices for standard classification tasks. The model has been trained by means of a 10 fold cross-validation procedure to choose the best number of trees T and their max depth D. The random split of the cross validation procedure is again performed at the fleet level, to avoid stops from the same fleet to be both in training and validation.

As a metric to evaluate each pair of parameters, we chose the widely adopted Area Under the ROC, or Receiver Operation Characteristic, curve (AUC for brevity). Such metric considers the curve of variation of the false positive rate (fpr) vs. the true positive rate (tpr) at different values of the classification threshold (ROC curve), where:

$$tpr = \frac{TP}{TP + FN}, \qquad fpr = \frac{FP}{FP + TN} \tag{1}$$

and TP, FP, TN and FN are the true positives, false positives, true negatives and false negatives respectively. The area under the ROC curve depends on both false positives and true positives: this lets it penalize models which are representative but not discriminative and, thus, makes it robust even in the case of unbalanced datasets. In addition, due to the imbalance on the number of entries in our dataset across all the fleets, to evaluate the performance during the cross-validation procedure we use the AUC median value on the 10 splits, rather than the mean, to avoid biasing our decision towards the fleets with the higher number of entries in the dataset. During the training procedure the best parameter values has been chosen varying $T \in \{100, 200\}$ and $D \in \{12, 15, 20, 22, 25\}$.

In the training phase, each group of features, as described in Section II-D, is progressively added. Each time we add a group of features, we train a new classifier with the optimal parameters found with a 10 fold cross-validation. This is done with the aim of evaluating the performance gain that we obtain with each of the groups. The sets of features used in the experiments are denoted as follows:

- `SWF`: stop-wise features;
- `CF`: cluster features;
- `POIF`: POI features;
- `SeqF`: sequential features.

The results of the cross validation procedure with the set of features increased progressively are discussed in Section III-C.

B. Baseline models

We compare our results with the method proposed in [13], that is, to the best of our knowledge, the only work on stop classification tackling a problem on a dataset with a large number of fleets spread across a wide geographic area (the whole Canada). As stated in the introduction, the authors base their classification on the diversity of fleets in an area, measured with fleet-wise entropy defined as:

$$E_f = -\sum_{f \in F} \frac{n_f}{N} \ln\left(\frac{n_f}{N}\right) \tag{2}$$

where F is the set of fleets present in the dataset, n_f the number of stops of the fleet f in the given area, and N is the total number of stops in that location. Rather than computing the entropy for each stop location, the entropy is computed partitioning the whole geographical area and assigning to each partition its entropy value. All the stops occurred in a given partition are assigned the entropy value of the partition, and the stops with highest entropy are classified as non-work order stops.

We adopt the same method to classify our dataset, adapting the geographical partitioning method to our data. We first divide the area with a grid on latitude and longitude defined with a step of 0.0025 degrees (which at the USA latitude corresponds to almost 250 meters), and then, using data coming from 30k companies in North America (again in the period February 2015 - January 2016), we computed the entropy with Equation (2). Note that, although our labeled dataset comprises only the fleets of 98 companies, the overall entropy E_f for each geographical partition should be computed on as many fleets as possible to provide meaningful information. Indeed, if we were to restrict the computation only to the 98 in our dataset, that are scattered throughout the USA, co-occurrence of vehicle stops of different fleets in the same location would be very rare, and the majority of computed entropy values would be equal to 0. Summing up, this baseline method classifies a stop S_i as a work order if the fleet-wise entropy is smaller than a given threshold, i.e., $E_f(i) \leq E_f(th)$. In the following we denote this baseline method by `BLEntr`.

As a second baseline method, the results reported in [10] and especially [17] suggest that the stop duration is by far the most important feature in problems related to stop classification. We can observe in Figure 1 that also in our dataset the stop duration is a highly discriminative feature.

Although we cannot adapt directly the methods in [10] and [17], because they use a few additional features that are not available or meaningful in our data, we decided to use the stop duration to construct a simple baseline model to compare with. We used a Random Forest classifier, trained simply with the stop duration feature, and we denote this baseline by `BLDur`.

C. Experimental results

We first set out to evaluate the performance of each group of features. In Figure 2 we report the boxplot of the AUC score across the 10 cross-validation folds. To assess the statistical significance of each of the groups of features, we perform 3 pairwise Wilcoxon tests [19] on the results obtained adding progressively more groups of features. The improvements given by the introduction of each group of features are all statistically significant at an $\alpha = 0.05$ level (with Bonferroni correction for multiple tests), with respect to the results obtained with only the previous sets of features.

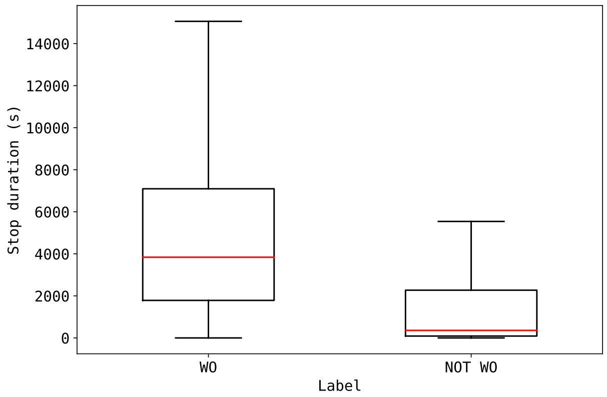

Fig. 1. Boxplots on the training set of the stop duration values for work order (WO) stops and not work order (NOT WO) stops.

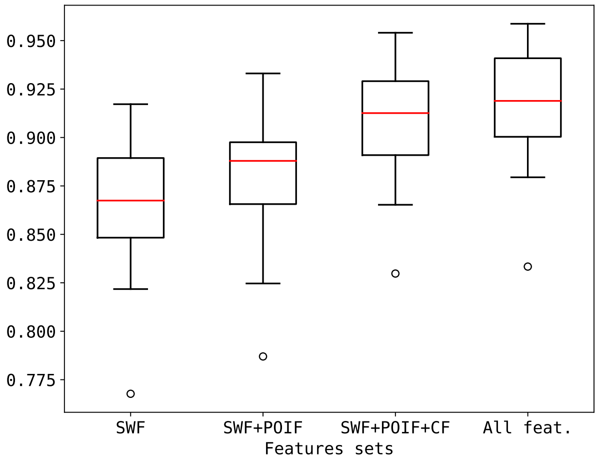

Fig. 2. Boxplots of the ROC AUC over the 10 folds of the cross-validation for each set of features. `All feat.` stands for: `SWF` + `POIF` +`CF` + `SeqF`.

In Table III we can see the performance of each model on the test set. The test results confirm the previous observation, obtained in cross-validation, that all the groups of features added to the dataset are beneficial in improving the classification performance significantly. The model trained with all the introduced features achieves an AUC score of more than 0.93.

The duration-based baseline method `BLDur` achieves a rather good performance, with an AUC of more than 0.85, again corroborating the idea that the stop duration is the main feature to be used in this problem. The entropy-based method `BLEntr` has a modest performance, 0.68, showing that it is not sufficient to correctly classify work stops. By looking at the bottom left of the ROC curve we can see that with high entropy values, the classifier is not able to distinguish between work and non-work stops.

All of our random forest models, regardless of the groups of features we included, outperform both baseline methods. For the sake of completeness, we also tested the inclusion of the fleet-level entropy E_f as a feature in the Random Forest

TABLE II

DATASET STRUCTURE.

Dataset split	Work order	Non-work order
Train	128071	396199
Test	33833	144343

TABLE III

PERFORMANCE OVER THE TEST SET AND OPTIMAL PARAMETERS OF THE CLASSIFICATION MODELS.

Model	Optimal parameters		ROC AUC
`BLEntr` - Baseline w. Entropy	–		0.678
`BLDur` - Baseline w. Stop Duration	$T = 100$	$D = 12$	0.854
`SWF`	$T = 100$	$D = 12$	0.878
`SWF` + `POIF`	$T = 100$	$D = 15$	0.890
`SWF` + `POIF` + `CF`	$T = 200$	$D = 20$	0.928
`SWF` + `POIF` + `CF` + `SeqF`	$T = 100$	$D = 22$	**0.931**

models, but it did not lead to an improvement. We do not include the result in the plots and tables for sake of readability.

In Table IV we show the top 3 features for each of the four groups of features along with their overall ranking and their relevance in the final model, trained with `SWF` + `POIF` + `CF` + `SeqF`. The reported scores have been obtained with an *a posteriori* feature ranking from the trained random forest, as suggested in [20], by accumulating for each feature the improvements in Gini impurity obtained in training by splitting on it. We can see that the features with most discriminative power are always related to the stop duration, taken either from the stop-wise, sequential or stop cluster sets of features. It is notable that the most important feature at all is a cluster feature, indicating that the characterization of the area where a stop occurs is crucial to correctly classify its purpose. Highly important is also the time with the engine off. Among the sequential features, time and distance from the previous and following stops appear to be useful. Interestingly among the POI features, only the distance from the closest fuel station, restaurant and vehicle repair facility have a noticeable impact on the classification.

Finally the performances of our best model on engine off and idling stops are evaluated separately, obtaining respectively 0.903 and 0.957 as ROC AUC value. We may think that the higher value for idling stops is due to the large amount of very short idling stops corresponding to traffic light and artifacts arising with heavy traffic conditions.

IV. CONCLUSIONS

This paper investigates the problem of stop purpose identification from GPS data of commercial vehicles. A method to extract stops aggregating raw GPS pings is introduced. To build a ground truth dataset, an automatic labeling procedure is implemented by looking at the work order schedule data for each vehicle taken into account. From the GPS data, we extract a rich set of features belonging to 4 different groups.

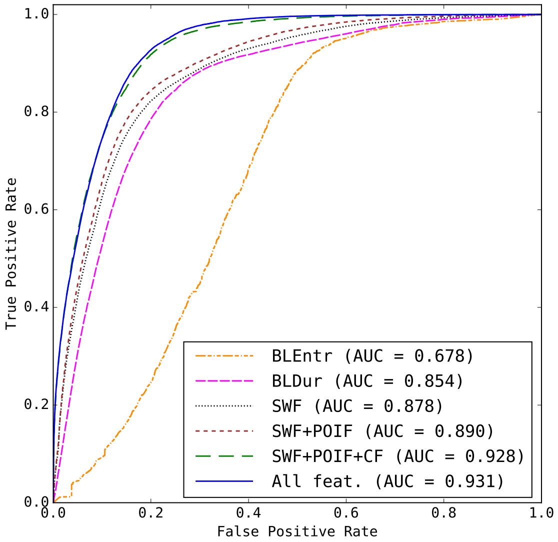

Fig. 3. Plots of the ROC curves of all the classifiers evaluated on the test set. `BLEntr` is the baseline model based on entropy, `BLDur` is the baseline model based on the stop duration. `SWF`, `POIF`, `CF` and `All feat.` refer to the groups of features used in the random forest model.

TABLE IV

FEATURE RANKING: TOP 3 FEATURES OF EACH GROUP WITH ITS OVERALL RANK AND SCORE.

Overall rank	Name	Feature set	Score
1	Average stop duration in cluster	CF	0.121
5	Max stop duration in cluster	CF	0.048
9	Sum of stop durations in cluster	CF	0.035
2	Stop duration	SWF	0.079
3	Total time with engine off	SWF	0.060
4	Max time with engine off	SWF	0.056
8	Time to next stop	SeqF	0.039
12	Distance from previous stop	SeqF	0.021
13	Distance to next stop	SeqF	0.020
19	Distance to closest fuel station	POIF	0.013
24	Distance to closest restaurant	POIF	0.008
29	Distance to closest vehicle repair facility	POIF	0.006

The first includes stop features extracted from each single stop, such as the stop duration, the number of pings in the stop, etc. A second group of features is comprised of information about the closest point of interest. Cluster features are also taken into account, computing statistics like the average stop duration in an area around a stop. Finally, sequential features are obtained by considering the sequence of stops of a vehicle in a single day.

The extracted features are used to train a Random Forest model, whose performance is compared with the one of two baseline models: the first is the entropy method introduced by [13]; the second is based on the duration of the stops, which is showed in the literature to be the most important feature in similar problems [17].

The experimental results show that the 4 sets of features significantly add classification power to the random forest. By looking at the feature ranking, it is easy to see that the features related to the duration of the stops are the most important ones, both computed for the single stop, and computed over cluster of nearby stops. Additional features, such as the time with the engine off, and the time to/from the neighboring stops, are also quite relevant. From the POI set of features, the distance from the closest fuel station is the most important for the classification, followed by restaurants and vehicle repair facilities. All the features groups combined together provide a highly predictive set of 100 heterogeneous features, letting our method outperform the two baselines in terms of area under the ROC curve. The best model achieves more than 0.93 in AUC score, compare to 0.85 of the best baseline.

Several future directions can be envisioned for this work. The dataset could be used to tackle multi-class classification, *e.g.* by enlarging the label set to include more specific types of stops, rather than the current binary labels (work order vs non-work order).

In addition, further sets of features could be included: for instance, we believe that different industry segments have different work order characteristics, then features indicating the industry type, if available, or some statistics to describe the average operations of the fleet could be relevant. Unfortunately, we believe that in this study the number of distinct fleets is not big enough to learn from this information.

Finally, stemming from the fact that information on previous and subsequent stops turned out to be quite effective, but perhaps less than expected, one could envision to further exploit the full sequence of stops by explicitly modeling the time component, e.g. with a graphical model or a recurrent neural network.

References

[1] G. Xiao, Z. Juan, and C. Zhang, "Travel mode detection based on GPS track data and Bayesian networks," *Computers, Environment and Urban Systems*, vol. 54, pp. 14–22, 2015.

[2] P. Gonzalez, J. Weinstein, S. Barbeau, M. Labrador, P. Winters, N. L. Georggi, and R. Perez, "Automating mode detection using neural networks and assisted GPS data collected using GPS-enabled mobile phones," in *15th World congress on intelligent transportation systems*, 2008.

[3] Y. Zheng, L. Liu, L. Wang, and X. Xie, "Learning transportation mode from raw GPS data for geographic applications on the web," in *Proceedings of the 17th international conference on World Wide Web*. ACM, 2008, pp. 247–256.

[4] A. Bolbol, T. Cheng, I. Tsapakis, and J. Haworth, "Inferring hybrid transportation modes from sparse GPS data using a moving window SVM classification," *Computers, Environment and Urban Systems*, vol. 36, no. 6, pp. 526 – 537, 2012, special Issue: Advances in Geocomputation.

[5] M. Simoncini, F. Sambo, L. Taccari, L. Bravi, S. Salti, and A. Lori, "Vehicle classification from low frequency GPS data," in *Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on*. IEEE, 2016, pp. 1159–1166.

[6] Z. Sun and X. Ban, "Vehicle classification using GPS data," *Transportation Research Part C: Emerging Technologies*, vol. 37, pp. 102–117, 12 2013.

[7] Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.-Y. Ma, "Mining user similarity based on location history," in *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*. ACM, 2008, p. 34.

[8] K. Jiang, H. Yin, P. Wang, and N. Yu, "Learning from contextual information of geo-tagged web photos to rank personalized tourism attractions," *Neurocomputing*, vol. 119, pp. 17–25, 2013.

[9] A. T. Palma, V. Bogorny, B. Kuijpers, and L. O. Alvares, "A clustering-based approach for discovering interesting places in trajectories," in *Proceedings of the 2008 ACM symposium on Applied computing*. ACM, 2008, pp. 863–868.

[10] L. Gong, H. Sato, T. Yamamoto, T. Miwa, and T. Morikawa, "Identification of activity stop locations in GPS trajectories by density-based clustering method combined with support vector machines," *Journal of Modern Transportation*, vol. 23, no. 3, pp. 202–213, 2015.

[11] M. Ester, H. peter Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise." AAAI Press, 1996, pp. 226–231.

[12] M. Lv, L. Chen, Z. Xu, Y. Li, and G. Chen, "The discovery of personally semantic places based on trajectory data mining," *Neurocomputing*, vol. 173, pp. 1142–1153, 2016.

[13] K. Gingerich, H. Maoh, and W. Anderson, "Classifying the purpose of stopped truck events: an application of entropy to GPS data," *Transportation Research Part C: Emerging Technologies*, vol. 64, pp. 17–27, 2016.

[14] F. Sambo, S. Salti, L. Bravi, M. Simoncini, L. Taccari, and A. Lori, "Integration of GPS and satellite images for detection and classification of fleet hotspots," in *20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017.

[15] J. Wolf, R. Guensler, and W. Bachman, "Elimination of the travel diary: Experiment to derive trip purpose from global positioning system travel data," *Transportation Research Record: Journal of the Transportation Research Board*, no. 1768, pp. 125–134, 2001.

[16] W. Bohte and K. Maat, "Deriving and validating trip purposes and travel modes for multi-day GPS-based travel surveys: A large-scale application in the netherlands," *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 3, pp. 285–297, 2009.

[17] X. Yang, Z. Sun, X. Ban, and J. Holguín-Veras, "Urban freight delivery stop identification with GPS data," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2411, pp. 55–61, 2014.

[18] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[19] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[20] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1.