Driving Habits Data: Location Privacy Implications and Solutions

Jacob Bellatti[1], Andrew Brunner[1], Joseph Lewis III[1], Prasad Annadata[1], Wisam Eltarjaman[1], Rinku Dewri[1,*], and Ramakrishna Thurimella[1]

[1]Colorado Research Institute for Security and Privacy, Department of Computer Science, University of Denver, Denver, CO 80210, USA
[*]rdewri@cs.du.edu

ABSTRACT

The collection of driving habits data is gaining momentum as vehicle telematics based solutions become popular in consumer markets such as auto-insurance and driver assistance services. Driving habits data includes features such as time of driving, speed, acceleration/deceleration patterns, distance traveled, braking practices, and others. Although revealing driving habits can give us access to a number of innovative products, they also introduce a privacy risk that consumers ought to be aware of. Using speed and time data from real world driving trips, we show that the destinations of trips may be determined without having to record GPS coordinates. We address this concern by developing a proof-of-concept device that can provide insurers with the aggregate statistics they seek, but without the requirement to transmit the data points collected during a trip.

Introduction

Many auto-insurance customers are familiar with the insurance discounts one can get by enrolling in telematics-based pay-how-you-drive programs. Examples of such programs in North America and Europe include Progressive's Snapshot, AllState's Drivewise, State Farm's In-Drive, National General Insurance's Low-Mileage Discount, Travelers' Intellidrive, Esurance's Drivesense, Safeco's Rewind, Aviva's Drive, Amaguiz PAYD, Insure The Box, Coverbox, Ingenie, MyDrive, and others. These programs rely on the collection of *driving habits data* during a monitoring period, which is later analyzed to offer a customized discount to the enrollee. Vehicle telematics based programs offer many advantages to insurers and the consumers. Insurers can offer more accurate pricing to consumers based on their driving habits. This increases affordability for safe drivers, and motivates others to adopt safer driving habits. More accurate actuarial data available to insurers will lead to more accurate assessment of risks, and may lead to more targeted training programs or safety measures. The market share of these programs is steadily increasing and expanding throughout the globe, with an estimated $80 billion in premiums by 2020.[1]

Driving habits data includes features such as time of driving, speed, acceleration/deceleration patterns, distance traveled, braking practices, and others. Unless the associated service explicitly requires customer tracking, collection of location data is avoided for privacy concerns. Typical auto-insurance discount programs (propelled by driving habits data) are opt-in programs where the driver has to enroll to be evaluated for a discount in her insurance premium. Upon enrollment, the driver receives a data collection device that can be plugged into the on-board diagnostic (OBD) port of the vehicle. The device collects driving habits data over a period of several days to few months. Some devices, such as those used by the Progressive Snapshot program, can periodically upload the data to a background server using consumer telecommunication networks. The device is returned to the agency at the completion of the data collection phase. Based on factors such as distances driven, time when driven, absence of hard brakes, and others, the driver is issued a discount in the insurance premium for the current and future terms. While few programs disclose that their data collection devices track the driver, most do not (or at least claim not to) track GPS locations, and imply an expectation of privacy that the customer's destinations are not tracked. Consumer surveys have reported that large fractions of the consumer base are opening up to these discount programs, although significant fractions are concerned about insurers tracking their driving destinations and sharing the collected data with unknown third parties.

A number of researchers have shown that privacy cannot be guaranteed simply by avoiding sharing or avoiding the collection of private data. The possibility of linking using quasi-identifiers, or other sophisticated methods, always remain. Quasi-identifiers are attributes of a database record that are non-identifying by themselves, but can be used to uniquely identify individuals when used in combination. A classic example is the re-identification of Governor William Weld's health records from an anonymized data set, based on a combination of gender, postal code and date of birth.[2,3] Along similar lines, research has shown how individuals can be identified by their social network structures,[4] or by their familial structures.[5]

Location inference is a deduction about the geographic location of an event from other known facts. We focus on the inference problem in the context of driving habits data collected *with* the consent of the driver. The underlying question is whether driving habits data attributes, such as the speed of a vehicle, can reveal the locations driven to by consumers, and if so, can auto-insurance discount programs be deployed in a privacy-preserving manner. The former question has received limited research attention,[6,7] while this work is the first known attempt to address the latter. Encryption based solutions have been explored for situations where insurers use a GPS-enabled device to offer pay-as-you-drive programs.[8]

To answer the research question, we present a location inference attack that executes on real traces of driving habits data, and attempts to identify the destinations of the trips during which the data were collected. Our techniques extract quasi-identifying information such as traffic stops, driving speed and turns from the data, and match them to publicly available map information to determine potential destinations of a trip. We observe that a number of trips can indeed be geographically matched to their destinations using simple driving features. Although not a foolproof method, our study shows that the destinations of certain trips can be very easily identified, thereby raising concerns about current expectations of privacy set by the data collection agencies. As the next step, we develop a proof-of-concept device similar to the ones used in driving data collection; albeit, with a "privacy by design" objective. This prototype collects driving habits data and, instead of storing these data points for post-processing, computes aggregate statistics of interest in the device itself. We report results that indicate that such a device is low-cost, and is more than capable of delivering the decision making statistics to insurance agencies.

Results

We used a commodity tracking device (LandAirSea GPS Tracking Key) to collect the raw data pertinent to this study. This battery powered device logs detailed driving data such as vehicle speed and GPS position, once every second, which can be later extracted into a computer through a USB connection. Although the device collects the GPS location (useful for validation later), the only data fields used in the inference process are: *time stamp* (t), *driving speed* (s), and *distance traveled* (d). We introduce here the term "trip" to mean a subset of the collected data, signifying a drive from one point of interest (e.g. home, office, hospital, store, friend's home, etc.) to another. Each $\langle t, s, d \rangle$ tuple of a trip is a data point of the trip.

We kept the devices in our vehicles for a period of 15 days in order to collect data from regular home-office trips, occasional shopping trips, and visits to infrequent places. We also collected a few trips between random locations at varying distances. During these trips, normal driving habits were maintained. We use a total of 30 trips in this study. All trips are in the Denver, Colorado area, and includes home to work and work to home drives, visits to the airport, the downtown area, local grocery stores, school drop-offs, social visits, and others. Length of trips range from 1 mile to 25 miles, and spanned interstate highways, state highways, city roads and residential areas.

Our location inference method works under the hypothesis that the points where a driver stops (stop-points) during a trip can be used as a set of quasi-identifiers for the destination of the trip. Therefore, if the start location of the trip is known, we can search a map of the area for paths that begin at the start location, and have traffic stops at distances given by the stop-points. The assumption of a known start location is not unrealistic, since the data collectors are typically aware of the street address where the vehicle is parked overnight. Start locations in subsequent trips can be obtained from the destinations of previous trips. Unless the roadways in the area are very regular, it is expected that a relatively smaller number of paths will satisfy the constraint to match every stop-point. The end-points of these *candidate paths* are potential destinations of the trip. We employ a ranking process when multiple candidate paths are identified.

Inferring trip destinations. Inference correctness depends on factors such as the frequency of stops made by the driver at intersections (stop-points), extent of deviation from the shortest possible route between the source and the destination, ability to drive at speed limits, and the correctness of the map data. The results of the algorithm are as follows—trips: 30; failure: 12; successful: 18; actual path in top three: 16; actual path is rank 1: 11. Therefore, we have a success rate of 60%, and if successful, there is a 88% chance that the actual route is in the top three detected paths. The number of potential routes ranged between 4 and 196 across the trips. An interesting observation is that, even if the top ranked destination is not the actual one, they are usually very close (within 0.5 miles) to each other. Therefore, most of the time, the locality of the destination can be inferred accurately! We did not find a correlation between the number of candidate paths and the ranking performance.

Illustrative example. Fig. 1 shows five candidate paths identified for one of the trips. A total of 196 candidate paths were found for this trip. All candidate paths match the four stop-points of the trip (7.95 miles in length). Candidate path 118 is also the actual route taken during the trip. The last plot in the figure shows the end nodes (destinations) of all candidate paths. Irrespective of the large number of candidate paths identified for this trip, most destination nodes cluster around a small number of localities. This is worth noting, since only four stop-points are involved over a distance of 7.95 miles in this trip; yet the ways to match them to an actual map are quite limited!

Fig. 2 compares the speed profiles of the actual trip and that generated by our driving model for a given path. Our model generates the speed profile that is likely to be maintained by a typical driver when driving along the path. It is clear that the

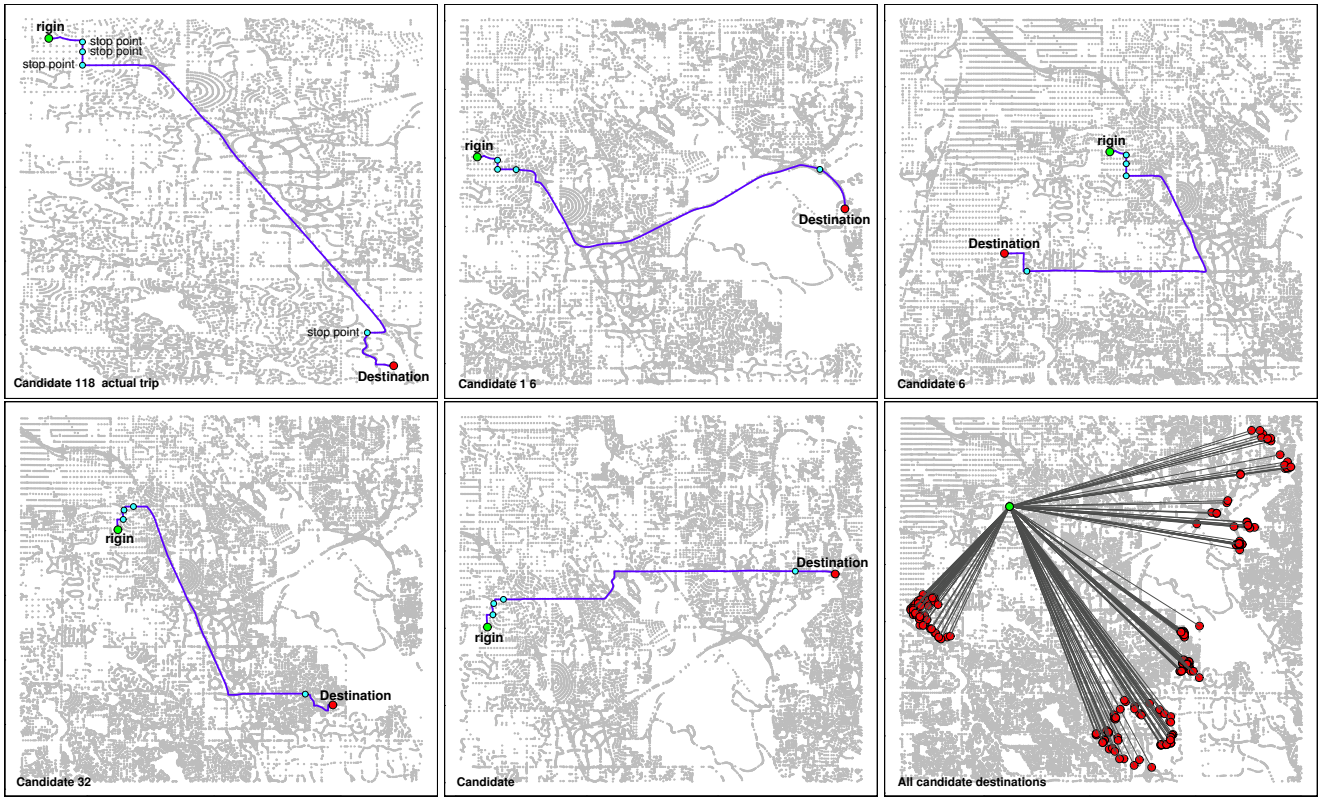

Figure 1. Sample candidate paths generated for a trip. Candidate path 118 is the actual route taken during the trip. The bottom right plot shows the destinations of all (196) candidate paths generated for this trip. Map data: ©OpenStreetMap contributors; licensed under ODbL.

more similar the speed limits and turns along a path are to that of the actual route, the higher is the ranking. Candidate paths 9, 32 and 118 progressively cover more of the highway, thereby increasing the match probability.

Private collection of driving statistics. Given the inference possibilities arising out of raw driving data collection, we built an aggregator device that continually collects and processes the raw data points and only stores higher level statistics corresponding to the data. This proof-of-concept device (Fig. 3) observes the speed of the vehicle once every second, utilizes it in computing various aggregate statistics, and then immediately discards the data point. The objective of the exercise was to determine if the observations of interest to a service provider can be computed in a low-cost device, as opposed to uploading the data to the provider for post processing. The device embeds a 16MHz ATmega328 microcontroller (see Methods). Using this low cost controller, we were able to keep track of the amount of time a vehicle is driven at different speed ranges (less than 30mph, 30–45mph, 45–55mph, more than 55mph, and more than 80mph), the distance driven, the number of stops made by the vehicle, and the number of hard accelerations and decelerations made by the vehicle. Processing of each data point took an average of 88 microseconds (minimum of 75 microseconds and maximum of 97 microseconds) during 55 random trips.

Discussion

The advantages of services that rely on the collection of driving habits data are noteworthy. Nonetheless, the threats of location tracking are equally concerning. Location tracking enables inferences about an individual's lifestyle[9–12] and social circles,[13] most of which may be considered private. Although the decision to share one's location is a personal one, such decisions can only be made when the intent to collect location data is fully disclosed. The difficulty arises when the location information is inferable from other types of seemingly unrelated data, in which case, either the possibility of inference is unknown to the business, or the location data is inferred and used without consumer consent.

The inference algorithm uses a probabilistic ranking method when multiple routes that match a trip's profile exist. The method is found to be robust in identifying the actual destination of a trip. If the destination is the end point of a candidate route, the route is often found in the three most likely paths that match the speed profile of the trip. The ranking procedure does a point-by-point probabilistic comparison of the speed values observed in the trip and that along an entire route. Therefore,

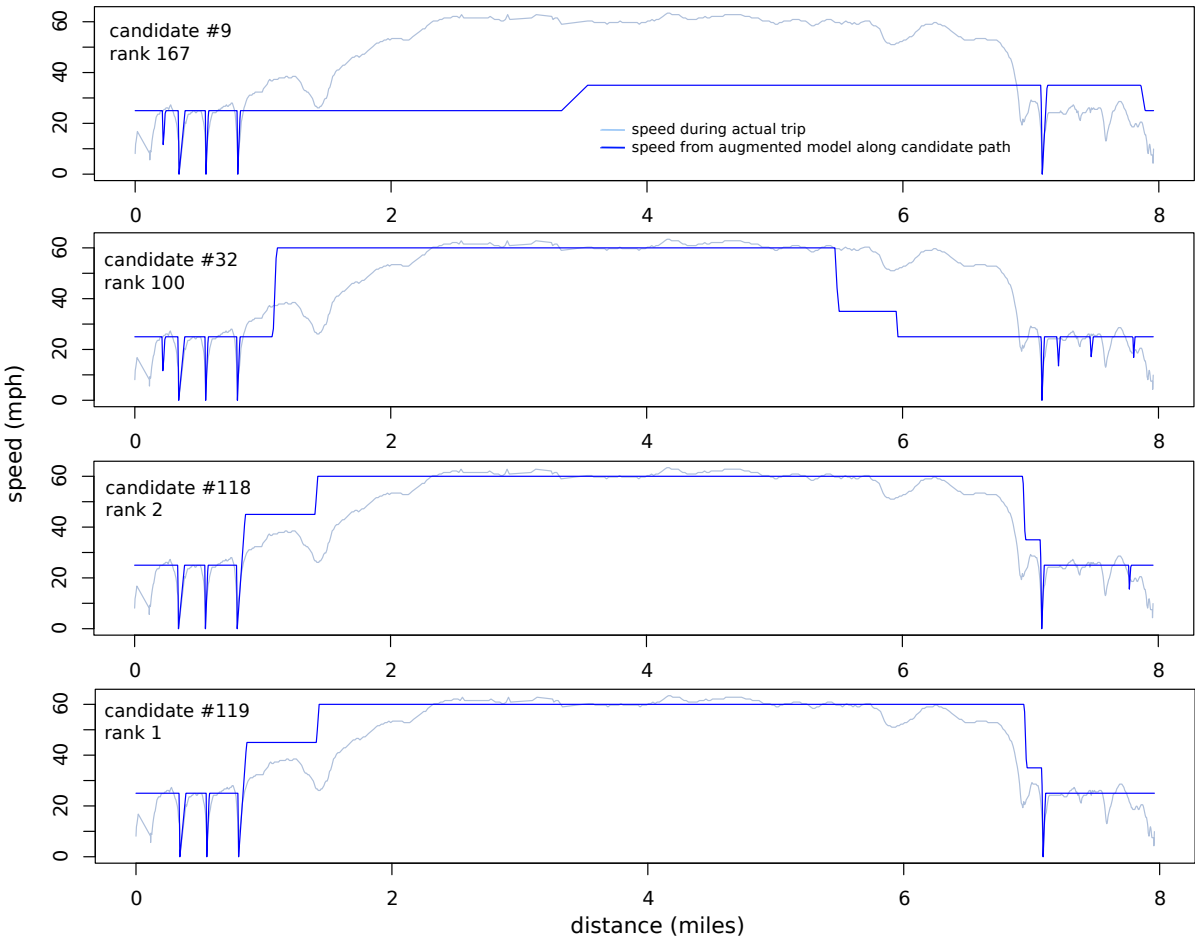

Figure 2. Speed profile during actual trip and that generated by a safe driver (called augmented model) for sample paths.

although we are not interested in the actual route followed during a trip, the obtained paths often represent the exact driving route. In addition, it is possible to cluster the destinations of the candidate trips to generate a rough probability map of where the trip ended. The ranking method suffers when speed limits are not reasonably followed, either due to excessive speeding or slow movement in traffic, and another candidate path matches this noisy speed profile.

We manually analyzed the 12 failed inferences to understand why a path to the actual destination was not discovered. For 4 out of the 12 paths, the trip involved a route that is not the shortest one (a factor that impacts the algorithm to some extent). For most others, a stop was made for a significantly long amount of time in the middle of the road due to heavy traffic. Our traffic pre-processing looks for more than one stop within a small distance; if a single stop is made due to heavy traffic, we will instead interpret it as a stop-point. In one case, the search was unsuccessful due to errors in the map data. The location inference algorithm presented here is driven by heuristics, which could be improved and redesigned for better accuracy. For example, Gao et. al. recently proposed another algorithm, called elastic pathing, to demonstrate the same privacy risks from driving habits data.[7] Elastic pathing uses an error metric to find candidate paths; the error is the difference between the observed distance travelled by a vehicle at a given point in time, and the distance if the vehicle was traveling on a given path. Lower error paths imply better matches, and should therefore be explored further. As more research pours into this problem, the quality of inference is likely to improve.

The aggregate statistics we chose for the privacy-preserving device are based on what many current insurance agencies seek to monitor in a driver's behavior. However, the processing times of the prototype indicate that multiple other statistics can be computed in the period between two data readings. Such statistics can include number of cornering attempts, amount of driving during peak and off-peak hours, partial speed snapshots when an event of interest occurs, average stopping distance, and overall health of the vehicle. The software in such devices can also be reconfigured to add or remove data processing functions. Also, since data points are not retained in the device, hardware interfaces capable of higher sampling rates can also be used for more accurate calculations. No storage overhead will be associated with a higher sampling rate hardware.

Figure 3. Privacy-preserving driving data aggregator device prototype.

From a consumer standpoint, a data collection device such as the prototype can be designed to allow users to view their statistics before they are transmitted to the insurance agency. Insurance discount programs based on driving behavior are often opt-in programs. The ability to a priori view the driving data that will be communicated to the insurance agency gives users the option to opt-out of the program at any given time. This can drive the trust between consumers and service providers, and insurance agencies can claim a program that is privacy-preserving by design.

Method

Area map as a graph. The first step to identifying candidate paths is to obtain a reliable map of the area. We obtained the map data available from the crowd-sourced OpenStreetMap project (wiki.openstreetmap.org). We processed the data files to generate a graph with 323928 nodes, and 639395 directed edges representing motorways, trunks, primary/ secondary/ tertiary/ residential roads, and corresponding link roads. Nodes are typically placed at intersections. Nodes are also placed between two intersections if the road in between is curved. Therefore, the length of a road segment can be accurately computed by aggregating the distances between successive nodes placed on the road segment. Each node is labeled with its latitude and longitude coordinates. Each edge is labeled with the geodesic distance between the two nodes of the edge. Distances are computed using the Vincenty inverse formula for ellipsoids. Edges are also annotated with a road type. This map data covers an area of more than 1500 sq. miles in Denver, Colorado and its suburbs, spanning between latitudes 39.41015^oN and 39.91424^oN, and longitudes 105.3150^oW and 104.3554^oW. We also assigned speed limit values to the edges of the graph. Since it was difficult to obtain the legal speed limit on all roadways, we assigned numbers based on the road type.

Data collection. We pre-process each trip to remove data points that may correspond to driving in traffic conditions. Our inference algorithm currently do not account for slow or "stop-and-go" driving resulting from heavy traffic; removal of data points collected during such conditions help infer locations accurately in more number of trips. Two steps are performed in this process. In the first step, we identify the data points where the driving speed is zero (possible stop in traffic). Thereafter, all data points between two zero-speed data points (inclusive) are removed if the total distance traveled between those two points is less than a threshold (half a mile used in this study). In the next step, consecutively time stamped zero-speed data points are removed if they do not span a time interval of at least 3 seconds. After the traffic pre-processing, we note the distance (unique) values corresponding to the remaining data points with a zero speed value. We refer to these distances as *stop-points*—possible distances from the beginning of trip where the driver had to halt due to traffic stops at signals and intersections. Traffic pre-processing can be made more accurate if traffic congestion data during the trip can be obtained from public sources.

Generating candidate paths. Candidate paths are generated by performing a standard depth-first search (DFS) of the map graph. The DFS starts at a node corresponding to the start location of a trip and outputs all paths that satisfy the list of constraints discussed next.

During the DFS traversal, we keep track of the length of the path from the start node. This constraint requires that, at any stage of the traversal, the current path must have an intersection node (3-way or more) at all stop-points less than the current length of the path. However, since traffic stops often happen a few feet away from the signal (the exact coordinates of the intersection), we allow for a *slack* while matching the path length to a stop-point. The slack is set to 500 feet in this study.

The second constraint requires that, at any stage of the traversal, a path to a node must always be the shortest one (within a slack of 0.1 miles) from the start node to that node. The constraint is motivated by typical driving behavior where a shortest path is preferred when traveling short distances inside the city. In such cases, shortest paths are often fastest paths too. This is

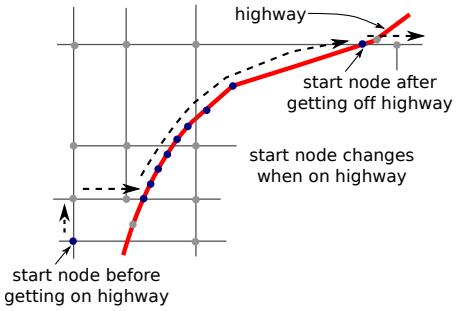

Figure 4. Disabling of shortest path constraint while exploring highway nodes.

a reasonable assumption in lieu of traffic conditions data at the time of the trip. However, the assumption fails when traveling long distances, where the driver is likely to take a faster (not necessarily shorter) route through the highway. Nonetheless, we can make the assumption that the driver would take the shortest route up to the highway, and then again from the point of exit on the highway to the destination. We incorporate this assumption by changing the start node to be the currently explored node, if the current node is part of a highway segment. As a result, the shortest path constraint remains disabled as long as the exploration continues on the highway nodes; the constraint is enabled when the exploration enters non-highway nodes, although the start node now is the last highway node (point of exit) on the path (Fig. 4).

The third constraint requires a path to always satisfy feasible speed limits at points of right and left turns. At every point of the exploration, we compute the angle by which a vehicle would have to turn when moving from the current node to the next node. An angle higher than 60^o is considered a turn, in which case we consult the trip data to ensure that the speed at that point of time was under 25 mph. We use the current length of the path to extract the closest data point from the trip, and use the speed in that data point as the current driving speed.

The fourth constraint terminates the exploration along a particular path when the path length exceeds the trip length. The path is then a candidate path if all stop-points (except the last one) have been matched in the path. When multiple candidate paths to the same end node are discovered, we retain the one with the least number of turns.

Candidate ranking. The DFS traversal for a given trip outputs the candidate paths that satisfy the four constraints discussed above. We process the candidates through a ranking procedure to arrive at the top inferred destinations of a trip. The ranking procedure makes use of information on typical speed limits along the candidate paths to find ones that best match the speed changes observed in the trip data points. We begin by first creating an ideal speed model for each candidate, then augment the model with driving behavior typically seen when making turns, and then compute a probability for the observed trip data to have been generated from the model. The candidates are ranked based on decreasing order of the probabilities.

The ideal speed model of a path P is a representation of the speeds that an ideal driver would follow when driving along the path under ideal conditions. An ideal driver is considered to be one who drives at exactly the speed limit, and ideal conditions imply no acceleration or decelerations in the driving speed. The model can be formally expressed as a function M of distance d and a path P. The output of such a function is the legal speed limit at distance d from the beginning of path P (assuming speed limit is same along both directions of travel).

$$M(d,P) = s^{limit}. \tag{1}$$

An ideal speed model can be improved by correcting the output speed in parts of the path where the vehicle would be performing a turn. Even an ideal driver in ideal conditions will decelerate to a reasonable speed to make a right or a left turn. A turn is assumed to happen exactly at the node joining the two edges that make the turn. We assume that all left turns happen at a speed of 15 mph and all right turns happen at 10 mph. The augmented model, denoted by M_{aug}, gradually reduces the output speed to the turning speed over a distance that depends on the acceleration and deceleration capabilities of the vehicle. Similarly, the model also incorporates the required acceleration behavior after the turn is complete. For all vehicles in this study, we use a fixed deceleration rate of 25 $feet/s^2$ ($= 7.8m/s^2 = 0.8g$, g being the acceleration of gravity), and a fixed acceleration rate of 6.5 $feet/s^2$ ($= 2m/s^2$). The augmented model also incorporates the information that the vehicle must have come to a complete halt at all stop-points. Similar to the turns, the output speed is corrected around the vicinity of the stop-points as well. Fig. 5 compares the speed values from a trip, and the values generated from the ideal speed model and the augmented model along a similar path to the same destination.

Given a trip $\mathscr{T}$ with n data points, $\langle t_i, d_i, s_i \rangle; i = 1, ..., n$, and a path P, we obtain the speed values generated by the

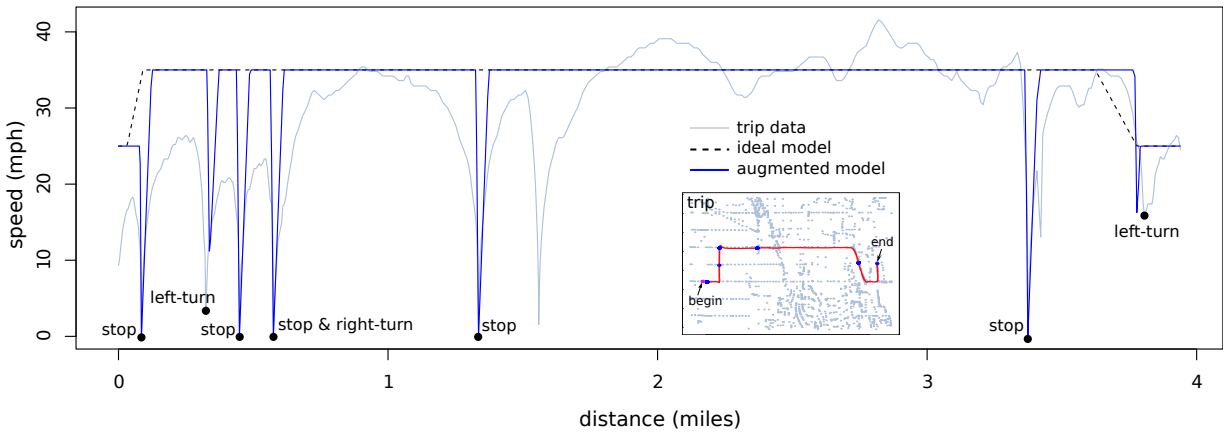

Figure 5. Speed profile for a trip, along with that generated from the ideal and the augmented models for a different path (differs in the first mile).

augmented model along path P at distances $d_1,...,d_n$. We denote these values by $s_1^{'},...,s_n^{'}$. The probability we seek is

$$Pr\left[\mathscr{T}|M_{aug}(d_i,P)=s_i^{'};i=1,...,n\right]. \tag{2}$$

We assume independence of speed values across time and distance, which gives us the probability as

$$\prod_{i=1}^{n} Pr\left[\langle t_i,d_i,s_i\rangle|M_{aug}(d_i,P)=s_i^{'}\right]. \tag{3}$$

Therefore, for each time instant t_i, we seek to compute the probability of observing speed s_i when the speed should have been $s_i^{'}$ at distance d_i along the path. The probability is computed from speed variation models based on standard Gaussian distributions. For speed value $s_i^{'}$, the density function used is

$$f=\begin{cases}\mathscr{N}(s_i^{'}+\frac{s_i^{'}}{10},\frac{s_i^{'}}{30}) & ,s_i^{'}\geq 20\text{mph}\\ \mathscr{N}(s_i^{'},1) & ,otherwise\end{cases}, \tag{4}$$

where $\mathscr{N}(\mu,\sigma)$ signifies a Gaussian density with mean μ and standard deviation σ. The distribution implies that, for speed limits of 20 mph or more, the mean driving speed is 10% higher, and 99.7% of the drivers drive between speeds of $s_i^{'}$ and $s_i^{'}+s_i^{'}/20$. For example, in a road with speed limit 60 mph, most drivers are assumed to drive at speeds between 60-72 mph, with 66 mph being the mean. For lower speed limits, we assume that drivers are more likely to stay close to the limit. The probability is then computed as

$$Pr\left[\langle t_i,d_i,s_i\rangle|M_{aug}(d_i,P)=s_i^{'}\right]=\int_{s_i-\varepsilon}^{s_i+\varepsilon} f(x)dx, \tag{5}$$

where ε is a negligible number (10^{-5}). To avoid issues of precision, we take the sum of the logarithm of the probabilities instead of the product of the probabilities at different time instances. The ranking is not affected because of this transformation.

Private data collection device. The above analysis is possible for a data collector when a vehicle's raw speed data is available. It is possible that this data collection is necessary because the collection device is not capable of processing the data points as they are obtained, and hence only buffers them and sends them later to the data collection agency for processing. To demonstrate otherwise, we explored a low cost device that attempts to process the data points in the device itself, and stores driving statistics that are claimed to be of value by insurance agencies. This is a "privacy by design" approach where the device does not hamper the business goals, and also does not leave open the possibility of location inferences such as the one above. Our prototype is based on an Arduino Nano board, which offers a 16 MHz ATmega328 microcontroller, 32KB of flash memory to store programs, 1KB of EEPROM (electrically programmable memory) for non-volatile storage, and can be powered by a vehicle's OBD-II port (Fig. 3). The software we inserted into this device reads the vehicle's speed every second, and streams the data point through various processing functions before discarding it. Each processing function keeps track

of one aggregate statistic. We choose ten statistics in this study, and record them for each trip: amount of time the vehicle is driven at speed ranges less than 30mph, 30–45mph, 45–55mph, more than 55mph, and more than 80mph; the distance driven; the number of stops made by the vehicle during a trip; the number of hard brakes (a deceleration of more than 7mph in a second) and steep accelerations (more than 7.5mph in a second) made by the vehicle; and the total trip duration. Using a basic encoding, the statistics from each trip are stored using 18 bytes in the device's non-volatile storage, allowing us to store up to 56 trip-statistics in the device before they need to be uploaded to the data collector. Based on the processing times we observed, this prototype device is capable of handling much faster rates of data collection, and many more processing functions.

References

1. Ptolemus Consulting Group. UBI Global Study, 8 Sep. 2015; www.ptolemus.com/ubi-study (2013).
2. Golle, P. Revisiting the Uniqueness of Simple Demographics in the US Population. *Proc. 5th ACM Workshop on Privacy in Electronic Society* (WPES 06), Alexandria, VA, USA. ACM. 77–80 (2006, October).
3. Sweeney, L. Weaving Technology and Policy Together to Maintain Confidentiality. *Journal of Law, Medicine and Ethics* **26**, 98–110 (1997).
4. Narayanan, A. & Shmatikov, V. De-Anonymizing Social Networks. *Proc. 2009 IEEE Symposium on Security and Privacy* (S&P 09), Oakland, CA, USA. IEEE. 173–187 (2009, May).
5. Malin, B. Re-identification of Familial Database Records. *AMIA Annual Symposium Proceedings*, Washington, DC, USA. AMIA. 524–528 (2006, November).
6. Dewri, R., Annadata, P., Eltarjaman, W. & Thurimella, R. Inferring Trip Destinations From Driving Habits Data. *Proc. 12th ACM Workshop on Privacy in the Electronic Society* (WPES 13), Berlin, Germany. ACM. 267–272 (2013, November).
7. Gao, X., Firner, B., Sugrim, S., Kaiser-Pendergrast, V., Yang, Y. & Lindqvist, J. Elastic Pathing: Your Speed is Enough to Track You. *Proc. 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (UbiComp 14), Seattle, WA, USA. ACM. 975–986 (2014, September).
8. Troncoso, C., Danezis, G., Kosta, E., Balasch, J. & Preneel, B. PriPAYD: Privacy-Friendly Pay-As-You-Drive Insurance. *IEEE Transactions on Dependable and Secure Computing* **8**, 742–755 (2011).
9. Farrahi, K. & Gatica-Perez, D. Discovering Routines from Large-Scale Human Locations Using Probabilistic Topic Models. *ACM Transactions on Intelligent Systems and Technology* **2**, Article 3 (2011).
10. Davidoff, D., Ziebart, B. D., Zimmerman, J. & Dey, A. K. Learning Patterns of Pick-ups and Drop-offs to Support Busy Family coordination. *Proc. SIGCHI Conference on Human Factors in Computing Systems* (CHI 11), Vancouver, BC, Canada. ACM. 1175–1184 (2011, May).
11. Krumm, J. & Brush, A. J. B. Learning Time-based Presence Probabilities. *Proc. 9th International Conference on Pervasive Computing* (PerCom 11), San Francisco, CA, USA. Springer. 79–96 (2011, June).
12. Zheng, Y., Chen, Y., Li, Q., Xie, X. & Ma, W.-Y. Understanding Transportation Modes Based on GPS Data for Web Applications. *ACM Transactions on the Web* **4**, Article 1 (2010).
13. Wang, D., Pedreschi, D., Song, C., Giannotti, F. & Barabasi, A.-L. Human Mobility, Social Ties, and Link prediction. *Proc. 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD 11), San Diego, CA, USA. ACM. 1100–1108 (2011, August).

Appendix: Notations

n	number of data points in a trip.
$\mathscr{T}$	trip data$=\{\langle t_1,d_1,s_1\rangle,...,\langle t_n,d_n,s_n\rangle\}$.
t_i	timestamp when i^{th} data point was recorded.
d_i	distance travelled between time t_1 and t_i.
s_i	speed of vehicle when i^{th} data point was recorded.
$Pr\left[\mathscr{T}\|M_{aug}(d_i,P)=s_i^{'};i=1,...,n\right]$	probability of recording the data points in $\mathscr{T}$ along a path P, given the speed values generated by the augmented model along the path.
$Pr\left[\langle t_i,d_i,s_i\rangle\|M_{aug}(d_i,P)=s_i^{'}\right]$	probability that the recorded speed is s_i at time t_i along a path P, given that the speed as per the augmented model is $s_i^{'}$.
$\mathscr{N}(\mu,\sigma)$	Gaussian distribution function with mean μ and standard deviation σ.