A Multi-task Learning Framework for Road Attribute Updating via Joint Analysis of Map Data and GPS Traces

Yifang Yin
Grab-NUS AI Lab, NUS
idsyin@nus.edu.sg

Jagannadan Varadarajan
GrabTaxi Holdings
jagan.varadarajan@grab.com

Guanfeng Wang
GrabTaxi Holdings
guanfeng.wang@grab.com

Xueou Wang
Grab-NUS AI Lab, NUS
idsws@nus.edu.sg

Dhruva Sahrawat
Grab-NUS AI Lab, NUS
idsdhru@nus.edu.sg

Roger Zimmermann
Grab-NUS AI Lab, NUS
dcsrz@nus.edu.sg

See-Kiong Ng
Grab-NUS AI Lab, NUS
seekiong@nus.edu.sg

ABSTRACT

The quality of a digital map is of utmost importance for geo-aware services. However, maintaining an accurate and up-to-date map is a highly challenging task that usually involves a substantial amount of manual work. To reduce the manual efforts, methods have been proposed to automatically derive road attributes by mining GPS traces. However, previous methods always modeled each road attribute separately based on intuitive hand-crafted features extracted from GPS traces. This observation motivates us to propose a machine learning based method to learn joint features not only from GPS traces but also from map data. To model the relations among the target road attributes, we extract low-level shared feature embeddings via multi-task learning, while still being able to generate task-specific fused representations by applying attention-based feature fusion. To model the relations between the target road attributes and other contextual information that is available from a digital map, we propose to leverage map tiles at road centers as visual features that capture the information of the surrounding geographic objects around the roads. We perform extensive experiments on the OpenStreetMap where state-of-the-art classification accuracy has been obtained compared to existing road attribute detection approaches.

CCS CONCEPTS

• **Information systems** → **Data mining**; **Web services**.

KEYWORDS

Road attributes, multi-task learning, GPS trajectories, digital maps

ACM Reference Format:
Yifang Yin, Jagannadan Varadarajan, Guanfeng Wang, Xueou Wang, Dhruva Sahrawat, Roger Zimmermann, and See-Kiong Ng. 2020. A Multi-task Learning Framework for Road Attribute Updating via Joint Analysis of Map Data and GPS Traces. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3366423.3380021

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380021

1 INTRODUCTION

The service of ride-hailing providers significantly relies on the quality of the underlying digital map. Incomplete map data such as a missing road or even a missing road attribute can lead to misleading routing decisions or inaccurate estimation of a driver's arrival time. However, the updating of both commercial and free maps still heavily relies on the manual annotations from human. The high cost results in maps with low completeness and inaccurate outdated data. Take the OpenStreetMap (OSM) [9] as an example, which provides the community a user-generated map of the world, its data completeness and accuracy vary significantly in different cities. For example, in Singapore, while most of the roads are annotated with the one-way or two-way tags, only about 68% and 23% of the roads are annotated with the number of lanes and the speed limit in the downtown area.

In the past decade, only a few efforts have been made on the automatic derivation of road attributes from vehicles' GPS traces, to reduce the high cost of manual annotations of map data [7, 21]. These methods first map GPS trajectories to road networks by applying map matching algorithms. Next, they extract features from the GPS trajectories that are mapped to each road segment, and model the road attribute detection as a multi-class classification problem. Interestingly, the relations between different road attributes are always neglected in the previous work as each road attribute is modeled based on an individual classifier such as the decision tree [22]. Moreover, the impact of existing map data on the detection of missing road attributes has also not been investigated yet in previous methods. Intuitively, the speed limit of a road is related to the road type and even the surrounding environment in the vicinity of that road. Thus, the existing map data can be of great importance for the inference of a missing road attribute.

To address the above issues, we present a multi-task learning framework for road attribute detection via joint analysis of map data and GPS traces. Figure 1 illustrates the overview of our proposed system. As the sensor readings in GPS traces are continuous that are difficult to be directly used as the input features to a classifier, we compute the distributions of location, bearing, and speed in GPS

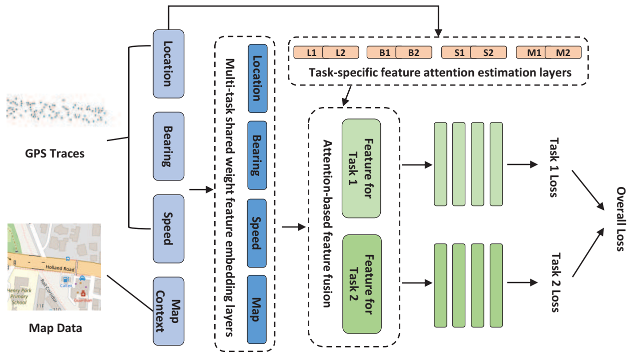


Figure 1: Overview of our proposed multi-task learning based road attribute detection.

traces that are associated with each road segment, and normalize the histograms using the L1 norm. Additionally, we extract features from the existing map by cropping a three-channel RGB image at the road center from map tiles. This image captures the contextual information around the road that can be very helpful for the missing road attribute detection.

Based on the observation that road attributes are always related to one another, we process the initial trace-based and map-based features by shared weight embedding layers to learn common feature embeddings among multiple tasks. The feature embeddings are next fused based on task-specific attention scores that are estimated by the attention module. This is based on the observation that the importance of a feature embedding is usually different for different tasks. For example, the location embedding can be more important to derive the number of lanes, and the speed embedding can be more important for the speed limit estimation. Finally, we model the detection of each road attribute as a multi-class classification problem, and optimize the neural networks by minimizing the overall loss that is defined as a weighted sum of the losses in each task. We summarize the key contributions of this work as follows:

- To the best of our knowledge, we are the first to investigate the use of the geographic information extracted from existing map data on the problem of road attribute detection.
- We present an effective multi-task learning framework for road attribute detection. The framework consists of three components: namely shared weight feature embedding layers, task-specific attention-based feature fusion layers, and task-specific classification layers.
- We conduct experiments on the OpenStreetMap data in Singapore. Our method is shown to significantly improve the classification accuracy, and overall provides new insights into the challenges in road attribute detection.

2 ROAD FEATURE EXTRACTION

We extract features to represent a road segment from two types of data sources: GPS traces and map data. The details of the feature extraction are introduced as below.

2.1 Feature Extraction from GPS Traces

A GPS trajectory is defined to be a sequence of records associated with timestamps. Each record consists of location, bearing, and speed returned by sensors. The location of a GPS record is usually represented by the latitude and longitude pair. The bearing is the clock-wise angle of the device's moving direction with respect to the earth's true north direction. As raw GPS traces are noisy and do not contain the information of the road segments on which they were travelling, we first perform the HMM-based map matching to find the group of traces that are associated with each road segment [17]. Formally, let $R = \{r_1, r_2, ..., r_n\}$ denote a set of road segments, and $P^i = \{p_1^i, p_2^i, ..., p_m^i\}$ denote the set of GPS points associated with road segment r_i where $p_j^i = (lat_j^i, lon_j^i, bearing_j^i, speed_j^i)$ is a 4-tuple that contains the readings of latitude, longitude, bearing, and speed. Based on P^i we extract the following three types of features for each road segment r_i from location, bearing, and speed, respectively.

Location Encoding. For each location $(lat_j^i, lon_j^i) \in P^i$, we compute the great circle distance between point (lat_j^i, lon_j^i) and road segment r_i [26]. As the distance is continuous in space that is infeasible to be directly used as a feature, we map the distance of 100 meters into 50 bins with each bin representing an interval of two meters [25]. Next we count the number of locations that fall into each bin and normalize the histogram using the L1 norm. We denote this feature as E_l. Intuitively, this feature is closely related to the number of lanes of a road segment.

Bearing Encoding. For each bearing $bearing_j^i \in P^i$, we compute the angle distance between the moving direction of the vehicle and the direction of the road segment r_i. We quantize the degree of 360 into 36 bins with each bin representing an interval of $10°$. Similarly, we count the number of bearings that fall into each bin and normalize the histogram using the L1 norm. We denote this feature as E_b. Intuitively, this feature is closely related to the road attribute of being one-way or two-way.

Speed Encoding. To encode the speed, we quantize the speed into slots where each slot denotes an interval of 10 m/s. We generate a histogram by counting the number of speeds that fall into each slot and normalize the histogram using the L1 norm. We denote this feature as E_s, which is intuitively related to the speed limit and average speed on each road segment.

2.2 Feature Extraction from Map Data

As aforementioned, the existing map data provides valuable contextual information that helps to derive the missing road attributes. But interestingly, the use of map data has not been investigated in previous methods. Figure 2 shows an example of the data representations from OpenStreetMap, which provides a free and user-generated map of the world. The map data is initially represented as key-value pairs. For example, $nodeid = 26782044$ indicates the unique node identifier, $lat = 1.2957609$ indicates the latitude of the node location, and $oneway = True$ indicates that this is a one-way road. This key-value pair representation is difficult to be directly used as features as the representation is inconsistent among different geographic objects with lots of missing, duplicate, or inaccurate values. Fortunately, as shown in Figure 2, every map has its own rules for

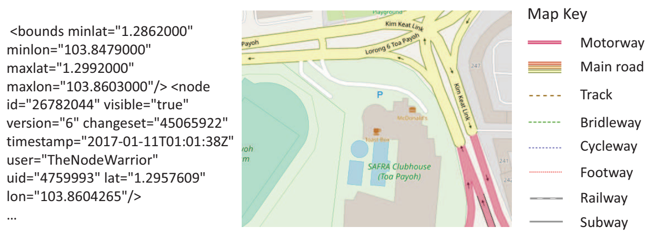
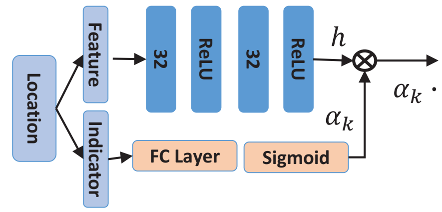
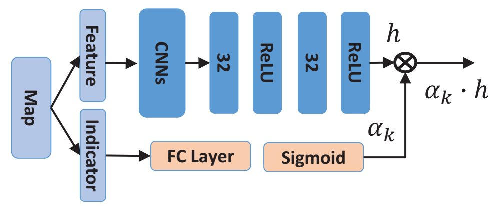


Figure 2: An example of the key-value pairs and the map visualization on OpenStreetMap.



(a) Sub-network for GPS traces processing.



(b) Sub-network for map data processing.

Figure 3: Feature embedding and attention estimation for features extracted from GPS traces and map data.

the visualization of the geographic objects including buildings and highways on a user interface. The different types and attributes of the objects are mostly differentiated based on color and line style. We thus propose to crop an image centered at each road segment from the map tiles. This image is considered as a visual feature, denoted as E_v, that captures the contextual information around a road for missing road attribute detection.

In addition to the aforementioned ability of handling the data inconsistency among the geographic objects from a single digital map, our proposed method can also easily integrate the information collected from multiple map sources. As the geographic objects (*e.g.*, roads) in different digital maps are usually identified by different IDs, it is challenging and time-consuming to align the information retrieved from different map sources. Graph-based algorithms [8] can be adopted to match the road networks, but the effectiveness of such algorithms can be easily affected by missing or inaccurate data from any of the digital maps. When dealing with multiple maps, our proposed method extracts images at road centers without actually aligning the geographic objects from different maps. We leave the job to each map to convert the raw key-value pairs into a three-channel RGB image, which is an intrinsic component for map visualization on a web-based user interface. We take the benefits of it by using the multiple images extracted from different maps as an additional visual input for road attribute detection. Convolutional Neural Networks (CNNs) will be adopted to extract features from the images. Small shift and misalignment in location among the images extracted from different map sources can be considered as data augmentation, which will have little (or even good) impacts on the training of the classifier.

It is also worth mentioning that instead of using a single image to represent the map data, it is also possible to generate a separate channel for each type of the geographic objects such as roads, buildings, rivers, *etc.* For example, Spruyt proposed to generate a 12-channel tensor from the map data to measure the semantic similarity between locations [20]. The advantage of using multi-channel tensor is that it reduces the information loss when converting map data into image tiles, but at the same time it increases the model complexity caused by the high dimensional input. We start with using a single RGB image as the feature extracted from the map in this work, and leave the investigation of multi-channel tensor representations on road attribute detection in the future.

3 ROAD ATTRIBUTE DETECTION VIA MULTI-TASK LEARNING

Previous methods process each of the road attributes separately. However, road attributes can be closely related to each other. For example, wide roads with more lanes can have a higher speed limit than narrow roads. Moreover, the availability of the ground-truth labels for different road attributes vary significantly in the map data. As aforementioned, while most of the roads are annotated with the one-way or two-way tags, only 23% of the roads are annotated with the speed limit. By learning multiple road attributes together, we are able to build a more diverse training dataset where the samples are only required to have (at least) one of the target road attribute labels. We therefore propose a multi-task learning framework [5, 14, 29] to address the issues of existing methods. The framework consists of the three components, namely the feature embedding layers, the attention-based feature fusion, and the task-specific classification layers.

3.1 Feature Embedding Layers

Multi-task learning has been shown to be effective in a variety of application by jointly analysing multiple tasks that are related to each other [12, 29]. In our framework, we adopt shared weight feature embedding layers to learn common patterns in the feature space among multiple tasks. Figure 3 illustrates the sub-networks for feature embedding and attention estimation. For features extracted from GPS traces, we process the initial embeddings E_l, E_b, and E_s by two fully-connected layers with 32 hidden units followed by the ReLU activation. For the image tiles extracted from the map, we first process the raw images E_v by a 2D CNN with three convolutional layers. We adopt a kernel size of 3 and set the number of filters to 32, 64, and 128, respectively. We apply 3×3 max pooling after each convolutional layer and pass the output of the CNN to two fully-connected layers with 32 hidden units followed by the ReLU activation. To reduce overfitting, we add a dropout layer after each fully-connected layer and set the drop rate to 0.3. The output feature vectors of the embedding layers, denoted as h^l, h^b, h^s, and h^v, will be fused based on task-specific attention scores and analysed by task-specific classifiers, the details of which are introduced as below.

3.2 Attention-based Feature Fusion Layers

We have investigated simple feature fusion techniques such as concatenation [18, 27] for the joint detection of multiple road attributes. However, such methods did not work well in this problem as the importance of the features extracted from different sensors vary significantly among different tasks. For example, bearing is closely related to one- or two-way detection, but is less correlated to the number of road-lanes. We thus propose to first estimate the importance of each feature in different tasks, and then fuse the features based on their importance rather than simply concatenating them together [16]. As shown in Figure 3, we estimate the feature importance based on the one-hot representation that indicates the feature type. Let $I^l = [1, 0, 0, 0]$, $I^b = [0, 1, 0, 0]$, $I^s = [0, 0, 1, 0]$, and $I^v = [0, 0, 0, 1]$ denote the one-hot indicators for the four types of features, respectively. We process the indicators by a fully-connected layer and the Sigmoid activation to generate task-specific feature attention scores. The number of hidden units in the fully-connected layer equals to the number of target tasks. We adopt the Sigmoid activation to ensure the attention scores to be in the range of $[0, 1]$. Let α_k^l, α_k^b, α_k^s, and α_k^v represent the attention scores (importance) of features E^l, E^b, E^s, and E^v in task k. The multimodal features are fused as,

$$h_k = \alpha_k^l \cdot h^l + \!\!\!+ \, \alpha_k^b \cdot h^b + \!\!\!+ \, \alpha_k^s \cdot h^s + \!\!\!+ \, \alpha_k^v \cdot h^v \qquad (1)$$

where $a + \!\!\!+ \, b$ represents the concatenation of two vectors. Though the shared-weight embedding layers generate shared global feature embeddings among different tasks, we are still able to learn task-specific fused representations h_k based on task-specific attention scores. This strategy has been shown to be more effective than feature concatenation with equal weights where the same fused representation is generated among different tasks.

3.3 Task-specific Classification Layers

For each task k, we make predictions based on the fused feature h_k by passing it to two fully-connected layers with 16 and 8 hidden units followed by the ReLU activation, and one output layer. To reduce overfitting, we add a dropout layer after each fully-connected layer and set the drop rate to 0.3. We model the detection of each road attribute as a multi-class classification problem and adopt the category cross entropy as the loss function. Let L_k denote the loss for task k, the final loss is defined to be,

$$L = \sum_k \beta_k \cdot L_k \qquad (2)$$

where β_k is in the range of $[0, 1]$, representing the weight of the loss for task k. We optimize the overall framework using the Adam optimizer with batch size set to 1024. The learning rate was set to 0.001.

4 EXPERIMENTS

We conducted experiments in three different areas in Singapore. The map data in these areas were retrieved from OpenStreetMap using a python library named OSMnx [4]. We target at three road attributes, namely one-way/two-way road, number of lanes, and speed limit, and derive the ground-truth labels from OSM data. We remove the road segments without ground-truth labels and divide the remaining into 80%-20% splits for training and testing. The number of training and testing samples in each task (*i.e.*, road attribute) is illustrated in Table 1. As can be seen, only bout 68% and 23% of the roads are labeled with road-lane numbers and speed limit, which again indicates the importance of automatic algorithms on missing road attribute detection. For feature extraction, we use the GPS trajectories of in-transit Grab drivers in Singapore [11] and the map tiles returned by the Map Tile API [10].

We compare the following methods and report the classification accuracy in Table 2.

- **DecTree** [22]: Design a decision tree for each road attribute separately based on manually crafted GPS features.
- **SinFea** and **SinFea-M**: Train a neural network for each road attribute separately based on a single feature only. SinFea uses the most relevant feature extracted from GPS traces, and SinFea-M uses the image extracted from map data.
- **ConFus**: Train a neural network for each road attribute separately based on multiple GPS features fused by simple concatenation.
- **AttFus**: Train a neural network for each road attribute separately based on multiple GPS features fused by task-specific attention scores.
- **AttMTL**: To model the relations between the road attributes, we presented a multi-task learning framework to jointly detect multiple road attributes based on GPS features fused by attention scores.
- **AttMTL-M**: To model the relations between road attributes and contextual information in existing maps, we cropped an image at each road center and fused it with features extracted from GPS traces in our proposed multi-task learning framework with attention-based feature fusion.

The DecTree method compared the heading of GPS points and the heading of the road, and clustered the points into three categories: "similar", "opposite", and "outliers". They adopted a threshold of 20° for the point clustering, removed the outliers, and computed the percentage of the number of points in the "similar" cluster. A road is considered to be one-way if the percentage is larger than 0.9. Though this method tried to remove outliers caused by intrinsic sensor noise, its performance can be significantly diminished by an improper threshold setting and inaccurate map matching noise. Moreover, this method failed to provide a solution for the number of lanes detection. The speed limit decision tree was also performed poorly on our dataset possibly due to the difference between countries and geographic regions.

The SinFea method trained a classifier based on a single, most relevant GPS feature for each task, *i.e.*, bearing for one/two way detection, location for number of lanes detection, and speed for speed limit detection. The SinFea-M method trained the classifiers using the image tiles extracted from map data. The results show that

Table 1: Numbers of samples in the training/testing datasets for the three road attributes.

Dataset	One/Two Way	No. of Lanes	Speed Limit
Area 1	1557/390	991/250	405/103
Area 2	2146/537	1501/379	389/101
Area 3	1205/302	848/217	328/84
Areas 1 & 2 & 3	4908/1229	3340/846	1122/288

Table 2: Classification accuracy comparison on road attribute detection of one/two way, number of lanes, and speed limit.

(a) Area 1

Method	One/Two Way	No. of Lanes	Speed Limit
DecTree	0.8692	-	-
SinFea	0.8846	0.4240	0.6506
SinFea-M	0.7205	0.4040	0.7864
ConFus	0.8897	0.4280	0.6796
AttFus	0.9026	0.4520	0.6893
AttMTL	0.9051	0.4640	0.7476
AttMTL-M	0.9154	0.5440	0.9029

(b) Area 2

Method	One/Two Way	No. of Lanes	Speed Limit
DecTree	0.8305	-	-
SinFea	0.8473	0.4987	0.7228
SinFea-M	0.7505	0.4802	0.8317
ConFus	0.8641	0.6332	0.7723
AttFus	0.8752	0.6332	0.7822
AttMTL	0.8827	0.6544	0.8218
AttMTL-M	0.9032	0.7177	0.8911

(c) Area 3

Method	One/Two Way	No. of Lanes	Speed Limit
DecTree	0.8940	-	-
SinFea	0.9139	0.5069	0.7143
SinFea-M	0.7483	0.4793	0.7976
ConFus	0.9073	0.5346	0.7381
AttFus	0.9272	0.5622	0.7619
AttMTL	0.9205	0.5899	0.7738
AttMTL-M	0.9205	0.6728	0.8690

(d) Areas 1 & 2 & 3

Method	One/Two Way	No. of Lanes	Speed Limit
DecTree	0.8584	-	-
SinFea	0.8804	0.5319	0.6285
SinFea-M	0.7469	0.4965	0.8160
ConFus	0.9064	0.5887	0.7604
AttFus	0.9089	0.6064	0.7812
AttMTL	0.9105	0.6052	0.7812
AttMIL-M	0.9211	0.6702	0.9028

the former is more effective for one/two way and number of lanes detection, while the latter is more effective for speed limit detection. This is related to the default map visualization for incomplete map data with missing key-value pairs. Next, we compare methods ConFus and AttFus, which fused the multiple features extracted from GPS traces based on concatenation and attention, respectively. Method AttFus outperformed method ConFus by 3% and 2% on the whole dataset in terms of the number of lanes detection and the speed limit detection, respectively. The results indicate that AttFus generally performs more effectively than ConFus, as AttFus assigns different attention scores to different GPS features when generating a fused representation.

Finally, method AttMTL improved method AttFus by applying multi-task learning. The results reported in Table 2 were obtained by assigning equal weights to the three tasks. On one hand, the shared-weight embedding layers in AttMTL learn global low-level features that are shared among multiple tasks. On the other hand, the attention-based fusion layers in AttMTL combine shared low-level features into task-specific fused representations for the prediction of each task. This strategy has been shown to be effective, especially on small to moderate datasets, with the following two advantages. First, it indicates that connections exist among different road attributes, thus improved classification results can be obtained by modeling the connections by multi-task learning. Second, it increases both the quantity and the diversity of training samples (especially for speed limit) as samples that are labeled with any one of the road attributes can be utilized to learn the shared low-level features among tasks. Finally, we present the AttMTL-M approach, which jointly analysed the features extracted from GPS traces and map data. As can be seen, the proposed method obtained the best road attribute detection accuracy among the seven methods. It outperformed the second best method AttMTL by 1.2%, 10.7%, and 15.6% for one/two way detection, number of lanes detection, and speed limit detection, respectively. The results thus demonstrate the effectiveness of our proposed approach.

Tables 3 and 4 report the per-class precision, recall, and F1 measure of methods AttMTL and AttMTL-M on number of lanes detection and speed limit detection. The results of ± one class are computed as follows. For a class c (*e.g.*, speed limit of 50 km/h), we retrieve all the samples with the predicted labels to be either c or the neighboring classes of c (*e.g.*, speed limit of 40 km/h and 60 km/h). We compute the recall of the retrieved samples for class c and report the results in column ± one class. This metric measures the "distance" between the prediction and the ground-truth label. For example, a high ± one class score for speed limit detection means that the predicted speed limit is close to the true speed limit of the road. Under such circumstances, the predicted road attributes can still be beneficial for downstream applications (*e.g.*, routing) without introducing significant errors. The number of test samples for the five classes in the road-lane detection is 132, 408, 169, 91, and 37, while that for the six classes in the speed limit detection is 20, 88, 151, 7, 17, and 5, respectively. Due to the problem of class

Table 3: Per-class precision, recall, and F1 measure comparison on the road attribute of number of lanes.

Class	AttMTL				AttMTL-M			
	precision	recall	F1	± one class	precision	recall	F1	± one class
1	0.7030	0.5379	0.6094	0.9697	0.6620	0.7121	0.6861	0.9470
2	0.7159	0.7966	0.7541	1.000	0.8430	0.7500	0.7938	0.9534
3	0.3986	0.6864	0.5043	0.9941	0.5775	0.6391	0.6067	0.9645
4	-	-	-	0.7912	0.3831	0.6484	0.4816	0.8791
5	-	-	-	-	-	-	-	0.8919

Table 4: Per-class precision, recall, and F1 measure comparison on the road attribute of speed limit.

Class	AttMTL				AttMTL-M			
	precision	recall	F1	± one class	precision	recall	F1	± one class
40	0.6316	0.6000	0.6154	1.0000	0.8235	0.7000	0.7568	1.0000
50	0.7284	0.6705	0.6982	1.0000	0.8587	0.8977	0.8778	1.0000
60	0.8354	0.9073	0.8698	1.0000	0.9351	0.9536	0.9443	1.0000
70	-	-	-	0.8571	1.0000	0.7143	0.8333	1.0000
80	0.7391	1.0000	0.8500	1.0000	0.8889	0.9412	0.9143	0.9412
90	-	-	-	0.6000	1.0000	0.4000	0.5714	0.6000

imbalance, it is more challenging to detect samples from the rare classes. We use "–" in the tables to represent that no instances from that class were detected and returned by the algorithm.

Generally speaking, method AttMTL-M is more robust as it outperformed method AttMTL in terms of the F1 measure in all classes. One advantage of AttMTL-M is that it performed more effectively in detecting samples from rare classes. Method AttMTL, on the other hand, tended to label samples as one of the major classes, resulting in obtaining relatively high recall and low precision compared to AttMTL-M in those classes. In terms of the ± one class measure, both methods obtained high recalls among the classes especially method AttMTL-M where most of the recalls it obtained were greater than 90%. It indicates that in most of the cases, the predicted class returned by our proposed method is either the true class or the neighbors of the true class. This measure can be an important indicator of the usability of the predicted road attributes in downstream applications, as it measures the level of errors introduced when annotating roads with the detected attributes.

5 RELATED WORK

A significant number of map inference algorithms have been proposed that aim at generating routable road maps from vehicle GPS traces [1, 15, 23]. Despite the efforts, most of the work concentrate on deriving the road network geometry [28] and topology [6, 19] while neglecting the detection of road attributes. Road attributes such as number of lanes and speed limit are of great importance as they can have a significant impact on routing decisions. As one of the pioneer work, Chen and Krumm presented a probabilistic model to derive the number of traffic lanes from GPS traces [7]. They proposed to use a Gaussian mixture model to model the distribution of GPS traces across multiple traffic lanes. Li *et al.* adopted the Support Vector Machine as the classifier to detect the road class and road name from a combination of movement trajectories and geotagged social media data [13]. Van *et al.* targeted at multiple road attributes, but they modeled each road attribute individually with a decision tree built on different features extracted from GPS traces [22]. Techniques that are tailored for a specific road attribute such as road type [3], road boundary [24], and lane detection [2] have also been proposed. However, none of the methods model the connections among multiple road attributes nor did they investigate the use of existing map data in the detection of missing road attributes.

6 CONCLUSION AND FUTURE WORK

Conventional road attribute detection methods extract intuitive hand-crafted features from GPS traces and model each road attribute separately. Thus to the best of our knowledge, we present the first multi-task learning based model for road attribute detection via joint analysis of GPS traces and map data. Our proposed framework models the relations among the road attributes via multi-task learning, which consists of three components: the feature embedding layers, the attention-based feature fusion, and the task-specific classification layers. The first component learns common patterns in the feature space among multiple tasks, which are next fused by the task-specific importance scores of the features computed in the second component. The third component predicts the attribute labels via task-specific classification layers, the losses of which are jointly minimized. Moreover, we extract contextual features from map data that contain the information of the geographic objects in the vicinity of a road, to facilitate the detection of missing road attributes. In the future, we plan to leverage a multi-channel tensor instead of a three-channel RGB image to model the contextual features extracted from the map. We can generate a separate channel for each type of the geographic objects, to reduce the information loss during the feature extraction.

ACKNOWLEDGMENT

This work was funded by the Grab-NUS AI Lab, a joint collaboration between GrabTaxi Holdings Pte. Ltd. and National University of Singapore.

REFERENCES

[1] Mahmuda Ahmed, Sophia Karagiorgou, Dieter Pfoser, and Carola Wenk. 2015. A Comparison and Evaluation of Map Construction Algorithms using Vehicle tracking data. *GeoInformatica* 19, 3 (2015), 601–632.

[2] Heba Aly, Anas Basalamah, and Moustafa Youssef. 2016. Robust and Ubiquitous Smartphone-based Lane Detection. *Pervasive and Mobile Computing* 26 (2016), 35–56.

[3] Heba Aly and Moustafa Youssef. 2015. semMatch: Road Semantics-based Accurate Map Matching for Challenging Positioning Data. In *ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 5:1–5:10.

[4] Geoff Boeing. 2017. OSMnx: New Methods for Acquiring, Constructing, Analyzing, and Visualizing Complex Street Networks. *Computers, Environment and Urban Systems* (2017), 126–139.

[5] Rich Caruana. 1997. Multitask Learning. *Machine Learning* 28, 1 (1997), 41–75.

[6] Chen Chen, Cewu Lu, Qixing Huang, Qiang Yang, Dimitrios Gunopulos, and Leonidas Guibas. 2016. City-Scale Map Creation and Updating Using GPS Collections. In *International Conference on Knowledge Discovery and Data Mining*. 1465–1474.

[7] Yihua Chen and John Krumm. 2010. Probabilistic Modeling of Traffic Lanes from GPS Traces. In *ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 81–88.

[8] S. Feizi, G. Quon, M. Mendoza, M. Medard, M. Kellis, and A. Jadbabaie. 2019. Spectral Alignment of Graphs. *IEEE Transactions on Network Science and Engineering* (2019).

[9] M. Haklay and P. Weber. 2008. OpenStreetMap: User-Generated Street Maps. *IEEE Pervasive Computing* (2008), 12–18.

[10] HERE Map. [n.d.]. https://developer.here.com/documentation/map-tile/topics/quick-start-map-tile.html/.

[11] Xiaocheng Huang, Yifang Yin, Simon Lim, Guanfeng Wang, Bo Hu, Jagannadan Varadarajan, Shaolin Zheng, Ajay Bulusu, and Roger Zimmermann. 2019. Grab-Posisi: An Extensive Real-Life GPS Trajectory Dataset in Southeast Asia. In *ACM SIGSPATIAL Workshop on Prediction of Human Mobility*.

[12] Keisuke Imoto, Masahiro Niitsuma, Ryosuke Yamanishi, and Yoichi Yamashita. 2019. Joint Analysis of Acoustic Event and Scene Based on Multitask Learning. *arXiv preprint arXiv:1904.12146* (2019).

[13] Jun Li, Qiming Qin, Jiawei Han, Lu-An Tang, and Kin Hou Lei. 2015. Mining Trajectory Data and Geotagged Data in Social Media for Road Map Inference. *Transactions in GIS* 19, 1 (2015), 1–18.

[14] Shikun Liu, Edward Johns, and Andrew J Davison. 2019. End-to-end Multi-task Learning with Attention. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1871–1880.

[15] Xuemei Liu, James Biagioni, Jakob Eriksson, Yin Wang, George Forman, and Yanmin Zhu. 2012. Mining Large-scale, Sparse GPS Traces for Map Inference: Comparison of Approaches. In *ACM SIGKDD*. 669–677.

[16] Xianglai Meng, Biao Leng, and Guanglu Song. 2017. A Multi-level Weighted Representation for Person Re-identification. In *Artificial Neural Networks and Machine Learning*, Alessandra Lintas, Stefano Rovetta, Paul F.M.J. Verschure, and Alessandro E.P. Villa (Eds.). 80–88.

[17] Paul Newson and John Krumm. 2009. Hidden Markov Map Matching Through Noise and Sparseness. In *ACM SIGSPATIAL*. 336–343.

[18] Andrew Owens and Alexei A Efros. 2018. Audio-visual Scene Analysis with Self-supervised Multisensory Features. In *European Conference on Computer Vision*. 631–648.

[19] Zhangqing Shan, Hao Wu, Weiwei Sun, and Baihua Zheng. 2015. COBWEB: A Robust Map Update System Using GPS Trajectories. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 927–937.

[20] Vincent Spruyt. 2018. Loc2Vec: Learning Location Embeddings with Triplet-loss Networks. https://www.sentiance.com/2018/05/03/loc2vec-learning-location-embeddings-w-triplet-loss-networks/.

[21] KBA Van Winden. 2014. Automatically Deriving and Updating attribute road data from movement trajectories. (2014).

[22] Karl Van Winden, Filip Biljecki, and Stefan Van der Spek. 2016. Automatic Update of Road Attributes by Mining GPS Tracks. *Transactions in GIS* 20, 5 (2016), 664–683.

[23] Jing Wang, Xiaoping Rui, Xianfeng Song, Xiangshuang Tan, Chaoliang Wang, and Venkatesh Raghavan. 2015. A Novel Approach for Generating Routable Road Maps from Vehicle GPS Traces. *International Journal of Geographical Information Science* 29, 1 (2015), 69–91.

[24] Wei Yang, Tinghua Ai, and Wei Lu. 2018. A Method for Extracting Road Boundary Information from Crowdsourcing Vehicle GPS Trajectories. *Sensors* 18, 4 (2018), 1261.

[25] Di Yao, Chao Zhang, Jianhui Huang, and Jingping Bi. 2017. SERM: A Recurrent Model for Next Location Prediction in Semantic Trajectories. In *ACM on Conference on Information and Knowledge Management*. 2411–2414.

[26] Yifang Yin, Rajiv Ratn Shah, and Roger Zimmermann. 2016. A General Feature-based Map Matching Framework with Trajectory Simplification. In *ACM SIGSPATIAL International Workshop on GeoStreaming*. 7:1–7:10.

[27] Yifang Yin, Rajiv Ratn Shah, and Roger Zimmermann. 2018. Learning and Fusing Multimodal Deep Features for Acoustic Scene Categorization. In *ACM International Conference on Multimedia*. 1892–1900.

[28] Lijuan Zhang, Frank Thiemann, and Monika Sester. 2010. Integration of GPS Traces with Road Map. In *Proceedings of the Third International Workshop on Computational Transportation Science*. 17–22.

[29] Yu Zhang, Ying Wei, and Qiang Yang. 2018. Learning to Multitask. In *Advances in Neural Information Processing Systems*. 5771–5782.